

EXPERT ADVICE ● TOP TIPS ● BEST BUYS ● NO GAMES!

WIN  
A CDTV!

# AMIGA

## SHOPPER

The top-selling serious Amiga magazine

ISSUE 11 ● MARCH 1992 ● £1.25 ● YOUR DEFINITIVE GUIDE

### 24-BIT PAINT

Why the Harlequin graphics card gives you more than just 16 million colours...page 22

### FRACTUALITY

Find out how to produce fractal animations with the impressive *Genesis*.....page 24

### STEREEEEEOOO!

Just how good is *Stereo Master*, the successor to *MasterSound*?.....page 31

### GET ON-LINE!

What you need and why it's worth joining the world of Amiga comms.....page 34

### EASY PRINTS

All you need to know to gain complete control over your printer .....page 82

### PD WORLD

Who on Earth is Fred Fish – and why does he have 580 PD disks for sale?...page 154



Just got an Amiga? Turn to page 69

## AMIGA ANSWERS

Got a problem? Pages  
- no problem! 41-71

## IS CDTV WORTH TAKING SERIOUSLY?

More software, lower price - but should you buy it? Read the definitive Shopper verdict

EXCLUSIVE: THE AMIGA'S FIRST  
VGA PC EMULATOR!  
see page 128

Future  
PUBLISHING











---

**ANNOUNCING  
THE**

**AMIGA**  
**SHOPPER**  
**SHOW**

**Wembley Exhibition Centre**  
**May 15 – 17**

**Seminars • New products**  
**Amazing bargains**

**DETAILS ON PAGE 125**

**You'll save serious money**





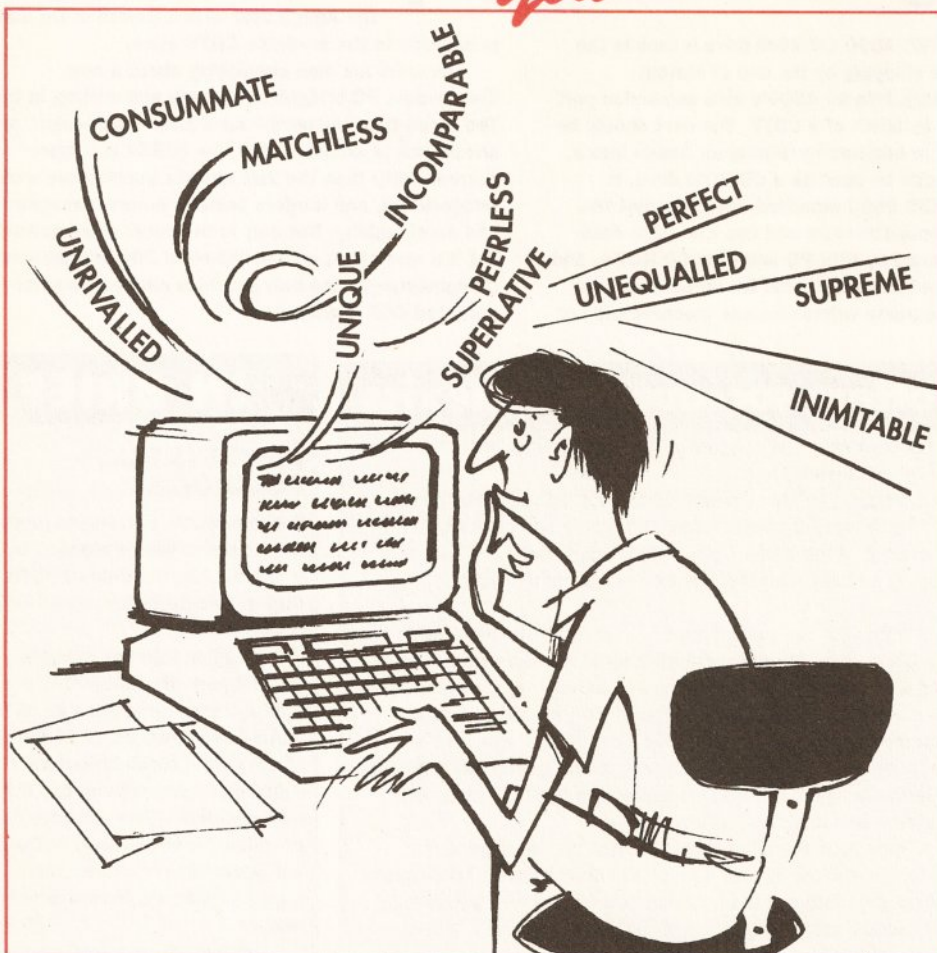






# PROTEXT 5.5

*"You won't be lost for words" . . .*



*"it's bloody brilliant"*

ST FORMAT

*"one hell of a performer"*

COMPUTER SHOPPER

*"if you need a professional word processor Protext is perfect"*

AMIGA COMPUTING

*"nothing else available comes close"*

ST APPLICATIONS

... with Protext 5.5 because the pop-up thesaurus will provide you with inspiration whenever you need it. With words provided by Collins the thesaurus has 43,000 main entries and 827,000 responses!

Protext 5.5 introduces enhanced text formatting options. Automatic hyphenation lets you produce a well-spaced page layout without the bother of manually putting in soft hyphens. Protext determines the correct hyphenation points by algorithms and look-up tables. Elimination of widows and orphans is also provided. You will no longer need to worry about those infuriating single lines at the top or bottom of pages. Protext formats the text to avoid these as you edit the text. Extra blank lines at the top of a page can be suppressed.

New document analysis features provide a wealth of information about your text. You can examine a list of all the words used - alphabetically or by the number of occurrences. Other statistics shown include average word length, average sentence length and a table of the number of lines on each page.

*Plus many more improvements . . .*

- ☆ Enhanced file selector with different sorting methods, bulk copy and erase.
- ☆ Prodata users - mail merge directly from Prodata files, no need to export.
- ☆ Mail merge: nested repeat loops.
- ☆ New window-based help facilities.
- ☆ Improved line drawing.
- ☆ Spelling checker finds repeated word and missing capital letters.
- ☆ Conversion to and from WordStar 5.5 and Microsoft RTF
- ☆ Full printed documentation of new features.

*Of course Protext still includes . . .*

Choice of pull-down menu or keyboard operation, extensive printer font support and proportional formatting while editing, up to 36 files open, split screen editing, characters for 30 languages, index and contents, footnotes, newspaper column printing, file sorting, macros, indent tabs, mail merge programming language, exec files and the fastest search and replace around. Altogether the most comprehensive word processing software for your Amiga or ST.

## PRODATA 1.2

New version of Prodata now with pull-down menus, mouse or keyboard operation, automatic record numbering, merge database, instantaneous filtering, prologue form, edit fields in any order, 2-across label printing. Full details available from Arnor.

Price: £85+VAT, upgrade from v1.1 £30+VAT.

## PRICES (including VAT and delivery)

For Commodore Amiga, Atari ST or TT.  
 Protext 5.5 £152.75  
 Upgrade from 5.0 to 5.5 £30  
 Upgrade from 4.2/4.3 to 5.5 £60  
 Please return your original disks when upgrading  
 French or German spelling dictionary £35.25  
 When upgrading please return any extra spell checking dictionaries for a free update to the revised version.

☆ Both Protext 5.5 and Prodata require 1Mb of memory



**ARNOR**

Arnor Ltd (AS), 611 Lincoln Road, Peterborough PE1 3HA. Tel: (0733) 68909 Fax (0733) 67299



































A surprisingly unique shape and much more compact than most impact printers.



An amazing capacity to handle 130 column printing on A3 landscape and A2 portrait.



Astonishing versatility includes multipart printing, bar-code printing and multi-emulation.

## WHEN YOU LOOK AT THE NEW DL1200 THE ONLY THING THAT'S NOT SURPRISING IS THE NAME FUJITSU.



The added distinction of a colour printing facility.



Astonishingly high-quality 24 pin printing.



A remarkably wide variety of designer fonts available on memory cards.



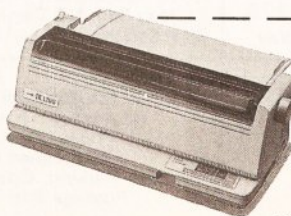
Incredibly, the DL1200 carries a price tag of under £500.



And all this courtesy of Fujitsu, the world's second largest computer company.

# FUJITSU

The global computer & communications company.



Please send me further information on Fujitsu's new DL1200 printer.

AS/3

Name

Title

Company

Address

Postcode

Phone  Fax

Fujitsu Europe Ltd, 2 Longwalk Road, Stockley Park, Uxbridge, Middx UB11 1AB. Tel: (081) 573 4444, Fax: (081) 573 2643.









































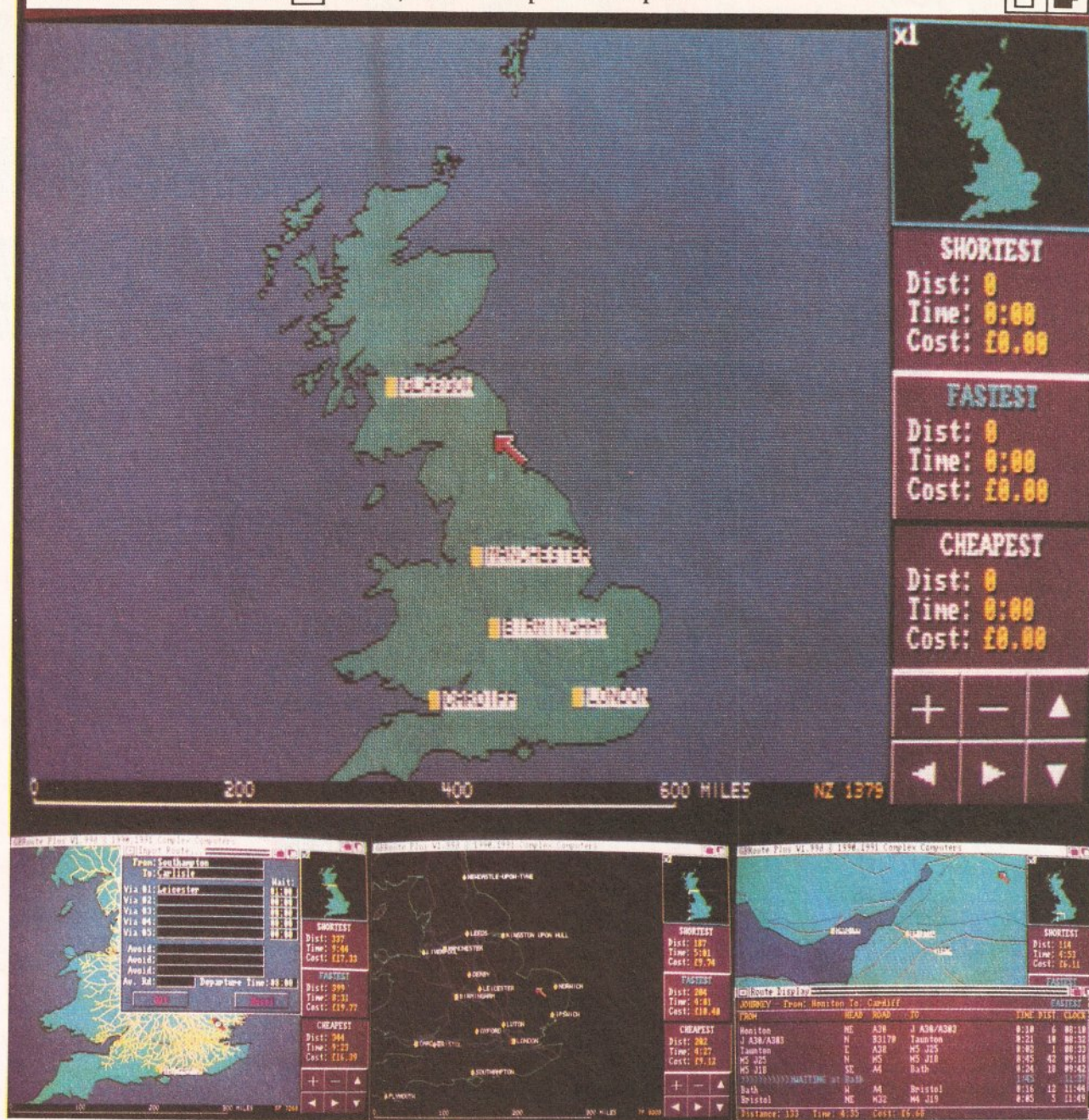












**£79.95** Inc Vat  
1 Meg Required

## GBRoute Plus

### Route Planner for the Amiga

**£79.95** Inc Vat  
1 Meg Required

GBRoute Plus for the Amiga computer has been carefully designed to assist the United Kingdom motorist plan easily and efficiently journeys almost anywhere in the UK. GBRoute Plus will not only find routes but will also cost, time and distance a journey in a matter of seconds, allowing the user to then print out a route or graphical map in either black and white or colour. There are many other features within the software designed for ease of use many of which allow the user to customise the program to suit the individual driver.

#### Features include:-

- \* Over 25 Million possible routes
- \* Calculate shortest, fastest and cheapest routes
- \* Includes Motorways, A roads and B roads
- \* Includes Cities, Towns and many Villages
- \* 5 Via options, 3 Avoid places and avoid road
- \* Colour and wireframe map graphics
- \* Interlace display
- \* Supports colour and black and white printers
- \* Map scrolling and zooming facilities

#### Features Include:-

- \* Motorway service stations included
- \* Road Bias facility for six classes of road
- \* 10 levels of magnification to a 1 mile scale
- \* Mileage scale bar
- \* National grid coordinates
- \* Colour palette requester
- \* ARexx support - over 50 commands
- \* Vector mileage calculator
- \* 3D look user interface

#### Dealers Contact:

1. Leisure Soft
2. HB Marketing
3. Laser Distribution
4. Centresoft/IBD
5. SDL

GBRoute Plus is available from most good Amiga dealers

Telephone: 0706 224531 - Fax: 0706 225320

Complex Computers-2 The Arcade, Waterfoot, Rossendale, Lancs BB4 9AF

















































































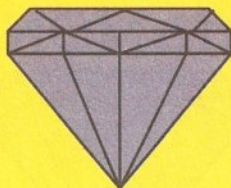








**WANT A 1500**  
With Workbench 2.04?  
**GOT A 500?**



**SWAP IT FOR**  
**ONLY £379.95**

UPGRADE  
YOUR OLD 500  
TO A BRAND NEW  
1500 FOR £379.95  
WHEN YOU P/X

# CAPTAIN DIAMOND'S 1500 & 3000 CENTRE

ALL PRICES INCLUDE VAT

AMIGA 1500 Specials A1500 GVP 52Mb + 8833 Mk II Monitor A1500 GVP 105Mb + 8833 Mk II Monitor <small>all MK II Monitors come with F19 promotion</small>	0Mb P/X		2Mb P/X		4Mb P/X		6Mb P/X		8Mb P/X	
	£839	£669	£899	£729	£959	£789	£1019	£849	£1079	£909
	£1039	£869	£1099	£929	£1159	£989	£1219	£1049	£1279	£1109
	£978	£808	£1038	£868	£1098	£928	£1158	£988	£1218	£1048
	£1178	£1008	£1238	£1068	£1298	£1128	£1358	£1188	£1418	£1248

PX= Part Exchange From Amiga 500 to Amiga 1500

**AMIGA 3000**  
The NEW  
Commodore AMIGA 3000  
25Mhz processor, 100Mb hard disk,  
4Mb Fast RAM, 2Mb Chip RAM,  
+ SCALA + Deluxe Paint IV  
**£3295.00** INC VAT

## AMIGA 1500 SPECIAL PROMOTION

### Comprising RHE NEW AMIGA 1500 PACK

Amiga 1500 52Mb 9ms access hard disk driven by the NEXUS SCSI hard disk controller with 2Mb of RAM which can be expanded in 2mb steps up to a maximum of 8Mb. The NEXUS also comes with a suite of storage software utilities: Flashback, Powerbench, SmartCache, Spoolit, DiskSurgeon, InstantFormat, MemoryDoctor. Utilities to enhance the control of your SCSI hard disks.

ALL DIAMOND 1500'S COME WITH NEW WORKBENCH 2.04

Software with the machine comprises DeluxePaint III, The Works, Home Accounts, 3 Games (Puzznic, Toki and Elf), and a book entitled "Get The Most Out Of Your AMIGA".

Or with 8833 MKII Monitor + F19 Promotion  
Or with Multisync Monitor and CBM Video Card (Flicker Fixer)

ONLY £899  
ONLY £1099  
ONLY £1349

## PROGRESSIVE PERIPHERALS

68040 Board for your A1500/2000  
50 MHz Clock speed

0Mb RAM **£1499**  
4Mb RAM **£1599**  
8Mb RAM **£1699**

## FUSION 40 68040

The Leader In Accelerator Cards  
with 4Mb on board RAM **£1299**

## GVP PRODUCTS FOR THE AMIGA 1500 & 2000

22MHz ACCELERATOR	33MHz ACCELERATOR	50MHz ACCELERATOR	IMPACT II HC8	150Mb STREAMER	IMPACT VISION 24
22MHz 68030 CPU 68682 22MHz maths co-pro 1Mb 32-bit RAM installed Expands to 13Mb 32-bit RAM. 32 bit SCSI HD on- board controller	33MHz 68030 CPU 68682 33MHz maths co-pro 4Mb 32-bit RAM installed Expands to 16Mb 32-bit RAM. 32 bit on-board SCSI controller	50MHz 68030 CPU 68682 50MHz maths co-pro 4Mb 32-bit RAM installed Expands to 32Mb 32-bit RAM 32 bit AT HD controller	Wide range of drive sizes 52Mb and 105Mb options Sockets for 8Mb RAM Uses easy to install SIMMs Supports virtually any SCSI device.	Over 6Mb per min. backup Comes with GVP TapeStore software Mounts in the 5.25" drive bay Optional external casing includes 1 DC150 cartridge	768x580 PAL resolution Supports composite video, S-VHS and RGB signals Broadcast quality genlock Software included
This low-cost, quality 68030 accelerator, memory upgrade, SCSI device and 68682 maths co-processor board offers the perfect introduction to low cost upgrades for the Amiga. It is supplied with 1Mb 32-bit RAM, upgradeable to 13Mb 32-bit RAM. This card runs approximately 7 times faster than a standard Amiga 1500/2000.	This fast 68030 accelerator, memory upgrade, SCSI device interface board delivers uncompromising speed and performance without a high price. It can take 16Mb 32-bit RAM and is supplied with 4Mb of 32-bit RAM installed along with a 68682 maths co-processor. Runs approximately 11.5 times faster than a standard Amiga 1500/2000.	The fastest accelerator card and memory upgrade available from GVP. Featuring a 68030 CPU running at 50MHz, on-card expansion to 32Mb of 32-bit RAM and an IDE hard disk interface all on one card. Makes your Amiga run approximately 13 times faster than a standard Amiga 1500/2000.	This is the equivalent of the HDB+ hard drive, but for the Amiga 1500 and 2000 models. It features the same unmatched facilities and performance giving unrivalled power. With GVP's innovative custom chip design, easy and user friendly installation and software, the HC8 is both powerful and simple to use.	This high-performance media back-up tape streamer device for the Amiga uses industry standard 150Mb DC150 cartridges. With backup speeds in excess of 6Mb per minute, the GVP WT150 is a fast and efficient back-up method to offer peace of mind for serious users. The WT150 tape streamer requires SCSI interface such as Impact II	This is the very latest 24-bit Professional Video Adapter. It features 16 million colours, built-in genlock, flicker fixer, frame buffer, frame grabber and digital keyer. Supplied with Caligari 3D modelling and rendering software, Scala IV24 video titling software, plus Macro Paint 24-bit paint package.
ONLY <b>£585</b>	ONLY <b>£1289</b>	ONLY <b>£1789</b>	52Mb version <b>£295</b> 105Mb version <b>£425</b>	ONLY <b>£739</b>	ONLY <b>£1785</b>

## SUPRA 8Mb BOARD FOR 1500/2000

Bareboard **£89**  
Add **£69** for 2Mb  
Add **£138** for 4Mb  
Add **£207** for 6Mb  
Add **£275** for 8Mb

## CAPTAIN DIAMOND 3000 "BASHER" PACK

The New Commodore Amiga A1500 with twin  
3.5" floppy drives, Multisync Colour Monitor,  
CBM Flicker Fixer (video card),  
68030 33MHz Processor, 105Mb hard drive  
with 4Mb RAM and s/w, Deluxe Paint III,  
The Works, Home Accounts  
+ 3 Games Puzznic, Toki and Elf.  
**ONLY £2595** INC VAT

## NEXUS BOARD

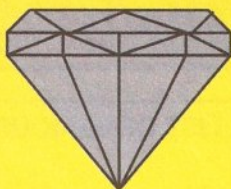
Bareboard 0Mb HD 0k RAM  
**ONLY £149**

Prices for Nexus Board options

RAM 0Mb 2Mb 4Mb 6Mb 8Mb  
No HD **£149 £199 £249 £299 £349**  
52Mb HD **£299 £349 £399 £449 £499**  
105MbHD **£399 £449 £499 £549 £599**



£449.95  
With Encyclopaedia  
& Lemmings



£479.95  
With Normal Titles  
& Starter Pack

## CAPTAIN DIAMOND'S CDTV PAGE

CDTV



CDTV

### Imagine, if you can, the concept of a new , more powerful Amiga...

...With 1Megabyte of memory and a compact disk drive of almost infinite size. A disk so vast, it can store hundreds of millions of digits of data. This data could be speech, animated pictures, digitised stereo sound, computer images or whole encyclopaedias... and more.

Imagine this and you can start to grasp the concept of CDTV.

The integral compact disk drive is the key to the power of CDTV. Its storage capacity is equivalent to around a quarter of a million full pages of text. This, when integrated with the 1Mb of internal RAM, creates a system, which, from a CD disk, can give you access to an unimaginable spectrum of real world images, never seen combined before. These vivid images, with words and sounds, give you a whole new dimension in home education, entertainment & information technology.

Don't forget that CDTV can play, in superb quality, any normal audio music CD. It can also play the new CD+G disks, which give digital sound and on-screen graphics.

On CD Disks now available, there are educational packages, encyclopaedias packed with reference information, stunning games, music systems and many other new and varied subjects including whole world atlases or even the complete works of Shakespeare, each one on CD Disk. CDTV - Nothing less than revolutionary

#### CDTV STARTER PACK

When you choose your CDTV from DIAMOND, not only do you get our legendary service, but for only an extra £30 you get the CDTV Starter pack to get you exploring your new world - straightaway. This includes a superb selection of CD Disk titles including the Welcome Tutorial & Hutchinsons Encyclopaedia and Lemmings plus four more games; worth in total almost £200. In addition is the infrared remote controller and all the hardware you need to get you connected.

#### CDTV with Starter Pack Only £479.95

CDTV is the same size and style as most video recorders, so it can sit unobtrusively above or below your home TV and/or HiFi. With its infrared controller it can be operated from your armchair. But, perhaps more importantly, if you are a computer enthusiast, remember that inside every CDTV there is an AMIGA just waiting to be used. So Later on you'll be able to buy the optional keyboard and disk drive to get you into the world of AMIGA computing. Printers, digitisers, genlocks and other accessories can all be connected to give access to many other exciting facilities and also many of the existing software packages available for the AMIGA computer. Never before has there been so much potential from one stunning system, packed with today's technology.

Call in to your nearest branch and have a personal demonstration, or phone for your fully detailed CDTV information pack.

#### CDTV ACCESSORIES

Due to be released soon  
Please call for availability.

<b>CD1220</b>	<b>Keyboard</b>	An 89 key QWERTY Keyboard	<b>£49.95</b>
<b>CD1252</b>	<b>Mouse</b>	Infrared mouse with battery saver	<b>£49.95</b>
<b>CD1200</b>	<b>Trackball</b>	Infrared trackball with the option of direct connection to extend battery life. Duplacates functions of two button mouse and includes two 9 pin ports for standard joysticks	<b>£79.95</b>
<b>CD1400</b>	<b>Caddy</b>	For holding CD within drive	<b>£9.95</b>
<b>CD1401</b>	<b>Memory Card</b>	Personal RAM card containing 64k of memory for storing data or as a bookmark facility within CDTV	<b>£79.95</b>
<b>CD1800</b>	<b>Floppy Drive</b>	Rack mountable floppy drive with 800k capacity. Casing allows room for additional hard drives or modems etc	<b>£99.95</b>
<b>CD1301</b>	<b>Genlock</b>	PAL based video interface card for superimposing CDTV images over a video signal provided by any PAL standard video source.	<b>£149.95</b>
<b>CD1321</b>	<b>Video Card</b>	PAL based video interface card to allow connection to TV's, videos etc requiring UHF, SVHS or composite inputs.	<b>P.O.A.</b>

**Matching External Disk Drive** **£64.95**

Many AMIGA accessories and peripherals (printers, disk drives, software etc) are compatible with CDTV.  
Please phone for information.

#### MULTIMEDIA IN YOUR OWN HOME

CDTV brings information and entertainment into your home in a way that has never been possible before. It is based around the familiar compact disc player combined with a computer which simply connects to your television and combines still and moving pictures with sound and thus offers a whole new world of possibility.

Whether you want to improve the quality of reference material available to your studies or to enhance your hobbies CDTV has the information at your fingertips. The clever design of the hardware and software means that you pick the information you want in the way that you want it.

Advanced Military	£22.47	Moving Gives Me		Shakespeare	
Fred Fish Collection	£37.47	A Stomach Ache	£26.25	Complete Works	£27.99
Snoopy	£26.25	Mud Puddle	£26.25	Illustrated Holy Bible	£27.99
Battlestorm	£29.99	Cinderella	£26.25	Music Maker	£23.99
American Vista	£41.25	Scary Poems Rotten Kids	£29.99	Barney Goes To School	£23.99
Sim City	£37.50	Long Hard Day At The Ranch	£26.25	My Paint	£27.99
Hound of the Baskervilles	£22.50	Wrath Of Demon	£22.50	A Bun For Barney	£23.99
Time Table Science	£29.99	Super Games Pack	£22.50	Defender Of Crown	£27.99
CD Remix	£22.50	Psycho Killer	£23.99	Garden Plants	£27.99
World Vista	£41.25	Classic Board Games	£26.24	Trees And Shrubs	£27.99
Eyes of the Eagle	£26.25	Time Table Business	£29.99	Fruit Veg And Herbs	£19.99
Peter Rabbit	£29.99	Science And Innovation	£31.99	Funschool 3(5-7)	£19.99
Benjamin Bunny	£29.99	Business Politics	£31.99	Mind Run	£23.99
Paper Bag Princess	£26.25	Dr. Wellman	£43.99	Timetable Of History	£31.99
Heather Hit Home Run	£26.25	Electric Cookbook	£31.99		
Thomas Snowsuit	£26.25	All Dogs Go To Heaven	£27.99		





# 68000 Assembly Language

*Our language column digs down even deeper as Paul Overaa has a look at assembly language and the Motorola 68000...*



"Assembly language is the best way to really get to grips with your Amiga - I'm here to tell you what it's all about."

Paul Overaa

humanized instruction names were called mnemonics because they were a memory aid which helped programmers to remember the purpose of the underlying processor instructions. The next step was to automate the process of converting mnemonics back to the numbers which represented the processor instructions. Programs which did this translation effectively 'assembled' the runnable program from the mnemonic instructions that the programmer had provided so they were called assemblers. And so, the assembly language was born!

Over the years, processors, assembly language programming concepts, and development software have all become increasingly sophisticated but these assembly languages (and each processor has its own) are always close to the actual machine and its underlying hardware - hence they are called low-level languages. The Amiga, as you'll already know, uses a microprocessor called the Motorola 68000 so to program at this level you need to learn 68000 assembly language!

## THE MOTOROLA 68000

The 68000 is a processor which has eight 32-bit data registers, seven 32-bit address registers, two 32-bit stack pointer registers, and a 32-bit program counter. There is also a 16-bit status register (divided into two eight bit registers) - these contain flag bits (the values of which are set and cleared according to particular results and processor operations), and a number of other mask/mode bits.

Five flags are available including ones for carry and overflow detection, and zero condition testing. If, for example, the result of some arithmetic operation was zero then the 68000's zero flag would be set. By convention the data registers are labelled d0-d7, the address registers A0-A6, and the combined twin stack pointer register A7. Even though the 68000 is a 16-bit processor the internal registers are 32 bits wide and many instructions can in fact work with bytes (8 bit), words (16 bit) or long word (32 bit) values. The assembly language programmer

identifies these variants by placing .b, .w or .l after instructions.

The 68000 instruction set is large and almost all sensible addressing modes can be used with any instruction. There's no way we can talk about, or even just list, each instruction - so here is a very brief outline of the type of things the 68000 can do:

## DATA MOVEMENT

The 68000 has a large number of instructions which allow the transfer of data to and from memory and/or the 68000 microprocessor's internal registers. Eg the instruction...

```
move.b d0, d1
```

transfers the lower eight bits of data from register d0 to register d1. This is an example of register addressing. On the other hand...

```
move.l #0, d1
```

places a zero value in register d1. The hash # sign indicates an operand source addressing mode known as 'immediate' addressing - in terms of the final 68000 instruction this means that the operand (in this case a 32-bit zero value) is stored immediately after the move.l instruction code.

Data can also be moved to memory locations, so to move the full 32-bit contents of register d0 to

**T**his month we take a look at assembly language - the programming language which is understood by the central processing unit and to which all other programs must be translated.

At the heart of any computer system is the processor. This will have registers for storing data, some hardware-oriented means of communicating with the outside world, and it will have its own 'instruction set', ie a collection of logic/arithmetic instructions which cause it to perform various tasks. The 'language' that the processor understands is based on binary numbers. Given suitable hardware (a processor chip, memory, some input/output facilities, and all the associated electronic support) one way of programming such a system would be to enter suitable binary numbers directly into system memory and then get the microprocessor to execute the instructions.

This approach was actually used to create and run programs in the early days of computing. It didn't take long before programmers realised that this sort of programming was a pain because the numbers which related to particular processor instructions didn't have any obvious connection with what the programmer was really trying to do. The solution was to give the instructions meaningful names (or as meaningful as possible) eg ADD, MOVE, SUB and so on. These

## Addressing modes

**O**ne of the most powerful features of the 68000 instruction set is the rich variety of addressing modes available. Most processor instructions work on a piece of data (the operand). This data has to be stored somewhere. Many instructions will use some real or implied source address, do something, and then transfer the result to its destination address - the processor's addressing modes enable these source and destination addresses to be specified. With the 68000 there are eleven basic addressing schemes and for completeness here are their names:

- Inherent
- Register
- Immediate
- Absolute
- Address register indirect
- Address register indirect with displacement
- Address register indirect with postincrement
- Address register indirect with predecrement
- Address register indirect with index and displacement
- Program counter relative with displacement
- Program counter relative with index and displacement

Inherent addressing means that the instruction itself implies the location of the operand. Register addressing implies that the operand resides in one of the 68000's internal registers.

Absolute addressing means that the ADDRESS of the operand is located just after the instruction in memory, whereas immediate addressing implies that the operand itself is located just after the instruction in memory.

Indirect addressing is a very powerful concept and on the 68000 a variant called register indirect addressing is used. In short, an address register is used to specify the ADDRESS of the operand. In addition to these straightforward addressing modes it is possible to specify displacements, to auto-increment or auto-decrement an address by 1, 2, or 4 bytes (this comes in handy for stepping through lists of 1, 2 and 4 byte data items) and to write program counter relative code (which is necessary when writing truly relocatable code).

It's not possible to explain all of these addressing modes (proper explanations would take an article in itself) but you'll find a couple of addressing mode notes elsewhere in this article.

a memory location which has been given the symbolic name `_DOSBase` you would use this instruction:

```
move.l d0, _DOSBase
```

## ARITHMETIC AND LOGIC INSTRUCTIONS

The 68000 supports a standard set of logic and arithmetic operations which allow it to perform addition, subtraction, multiplication and division. In addition to this it also supports all of the common logic operations (AND, OR, XOR etc.) As an example, the instruction:

```
add.l d0, d1
```

adds the full (32 bit) contents of data register d0 to the contents of register d1.

## FLOW CONTROL

Without flow control instructions a processor would only be able to execute program instructions sequentially. The ability to execute different parts of a program under different input/data conditions is fundamental to the nature of computing so the 68000, like all other processors, provides a number of useful mechanisms.

The 68000 provides both conditional and unconditional branch/jump type instructions for transferring control from one part of a program to another. One such instruction is called `beq` (Branch on Equal to zero) and this is a flow control branch which is only taken IF the 68000's zero flag is set. To use this instruction to conditionally branch to a symbolic address called `EXIT` you would write:

```
beq EXIT
```

Unconditional branch/jump instructions are also available and I'm always reminded, when I discuss this particular area, about Basic's `goto` instruction. This got the blame for helping programmers to produce tangled web, spaghetti type, programs which no-one could understand, debug or alter. `Goto` is now defunct within the world of high-level languages, discredited and argely unused. Any competent programmer, however, will tell you that `gotos` can be used properly and can result in tidy well structured programs. The difficulty is of course that it is only too easy to use the `goto` statement in an undisciplined way... and it's that which leads to program structure problems.

## THE GOTO INSTRUCTION

Why have I mentioned the `goto` at this time? It's because it has a strong connection with the branch and jump instructions of the 68000

processor. Programming at low-level then has all the disadvantages, yet none of the advantages, of the primitive high-level language facilities which have long since been superseded by forms which encourage the programmer to produce, or at least facilitate the production of, tidier programs. When you program using 68000 assembly language, or any assembly language, you'll find no such encouragement. To a large extent any structure and tidiness in the code will have to come from you the programmer.

Subroutine oriented branch and jump instructions are also available on the 68000 and these automatically store a 'return' address on the stack (after a subroutine call has been executed this return address is used to transfer control back to the main part of the program).

## OTHER INSTRUCTIONS

Instructions are provided which allow the 68000 to test, set, and clear individual bits and to rotate and shift operands. There are powerful address calculation instructions,

automated loop instructions, and even instructions which allow data areas to be allocated within stack space as subroutine calls are made. A variety of instructions are also available for comparing particular operand values (these set the appropriate status register flags).

There is of course a lot more to 68000 assembler programming than we've been able to cover in this article, but hopefully the general flavour of this style of low-level language programming will have become apparent. The instruction sets of most processors, even powerful ones like the 68000 used in the Amiga, are then quite limited and there is nothing complex about their operations. Each instruction carries out some elementary task, perhaps adding two values together or copying the contents of one memory location to another.

Despite this underlying simplicity there's no doubt that tackling 68000 assembly language is not a task to be undertaken lightly. Problems will arise when you try to work out how to combine thousands of assembly language instructions into a program

which does a particular job. It is a task which is error prone and time consuming. The benefits? Firstly you'll be able to make your programs run at the ultimate speed. Secondly, you will develop a 'gut feeling' for what computing is all about at the 'nuts and bolts' level.

## LAST WORDS

Assembly language programming on the Amiga adds another dimension – the complexity of the operating system itself. Before you can comfortably write assembler code to do a job you need to know enough about the operating system and its library code system call arrangements to work out what your assembler code should be doing.

Don't forget that it is often possible to combine high-level and low-level approaches. Here the bulk of the code is written as normal using a high-level language, and any routines which are critical are added as assembler patches. This gives the programmer the best of both worlds – high-level development coupled with absolute speed and control in the sections where it counts! **AS**

# A Short Example

For reasons of space it's not possible to give a fully fledged 68000 Amiga program but this example, though useless in terms of what it does, should give you an idea of what 68000 programming is all about. First of all here is the piece of code itself:

```
OPEN_DOS:
    move.l #dos_name,a1
    load pointer to library name
    move.l #0,d0
    place library version in d0
    CALLEXEC OpenLibrary
    this is an Amiga system
macro
    move.l d0, _DOSBase
    save the returned pointer
    beq EXIT test to see if it is non-zero
    CALLEXEC CloseLibrary
    only close if OpenLibrary was OK
EXIT:    rts logical
end of program
_DOSBASE ds.l 1
dos_name dc.b 'dos.library',NULL
```

An assembly language program consists, in the main, of statements which can contain up to four fields... a label, a mnemonic code, operands/addresses, and comments.

Labels are used to identify particular places in the program (ie they provide symbolic names which make the program more readable), the mnemonic codes and

operands/addresses fields are the instructions we've talked about and the comments are used to provide in-line program documentation.

Most of the instructions in the above example program have already been explained. The first statement, for instance, loads register a1 with the start address of the DOS library name (this name has been stored as part of the program's static data by using an assembler directive called `dc.b` (define byte constants)).

The next instruction places a zero in register d0. Why is this done? It's an Amiga system convention – we are using an Exec system call named `OpenLibrary` and this function (which is documented in the Amiga RKM manuals) requires the start of the library name (ie the address of its first byte) to be in register a1 and the version number to be in register d0!

The third program line, `CALLEXEC OpenLibrary`, is not a 68000 instruction – it is an assembler pseudo-instruction which identifies a group of instructions (called a 'macro'). In this particular example the real instructions are defined in the Amiga header files and the assembler inserts the appropriate instructions automatically.

The `OpenLibrary` system call, like a great many Amiga system

calls, may not succeed so when the program runs two possibilities present themselves:

**1:** The `OpenLibrary` call succeeds – in which case register d0 will contain a valid 'base address' for the library (this return value again stems from RKM documented system conventions).

**2:** The `OpenLibrary` call fails – here system conventions dictate that on return from such a call d0 will contain a zero failure indicator.

The program must take account of these eventualities which is what the example program does: the program stores, using a `move.l` instruction, the contents of d0 in a memory location which has been given the symbolic name `_DOSBase`. As the data is moved the zero flag is modified to reflect the value of the data item – a `beq` (Branch on Equal to zero) instruction is then used to decide whether or not the library was successfully opened. The result of this branch based jiggery-pokery is that the library is closed ONLY IF the library was successfully opened in the first place. The last program statement, an `rts` instruction, is the 68000's return-from-subroutine statement – in this case is being used to signify the end of the program.

# Media Direct 0782 208228

## AMIGA 500 PACKS



All our Amiga's are UK specification and come complete with the following:—

### Amiga Pack 1 (Classic Pack)

512K random access memory  
Built-in disk drive  
4096 colour graphics  
4 channel stereo sound  
Kickstart 1.3  
Workbench 1.3  
Extras 1.3 and tutorial disk  
Amiga mouse  
TV modulator  
Power supply unit  
A500 keyboard

Bart Simpson  
Captain Planet  
Deluxe Paint III  
and 'game of the year'  
Lemmings

**We also give you:—**  
Mouse mat and pocket  
dust cover, 10 blank disks  
and box, Quickshot  
Python, 1 joystick.

**Plus**  
10 more Great Games  
Football Manager  
Hotshot  
Las Vegas  
Flight Path 737  
Fireblaster  
Karting Grand Prix  
Thai Boxing  
Seconds Out  
Mouse Trap  
Plutos

### Amiga Pack 2 (Classic Pack Plus)

As Pack 1 but also  
includes the Cumana  
CAX354 external disk  
drive

£454.95

### Amiga Pack 3 (Classic Printer Pack)

As Pack 1 but includes the  
Star LC200 full colour  
printer

£604.95

### Amiga Pack 4 (The Ultimate Pack)

As Pack 1 but includes  
Star LC200 colour printer  
plus Philips CM8833/II  
stereo colour monitor

£834.95

### Amiga Pack 5 (Educational Pack)

As Pack 1 but includes a  
selection of educational  
software:—  
Spell Book (ages 4-9)  
Things to do with words  
(ages 5-12)  
Things to do with numbers  
(ages 5-10)  
Let's spell at the shops  
(ages 4-10)  
Let's spell at home  
(ages 4-10)  
Let's spell out and about  
(ages 4-10)  
Puzzle Book 1 (5 to adult)  
Puzzle Book 2 (5 to adult)

£499.95

### Amiga Pack 6 (Gamers Pack)

As Pack 1 but also  
includes some of the  
hottest games:—  
F29 Retaliator,  
F19 Stealth Fighter,  
Lotus Esprit Turbo  
Challenge,  
Swiv (Silkworm II),  
Flight of the Intruder.

£479.95

## PRINTERS

All our printers are UK  
specification and are  
covered by a 12 months  
warranty. All the prices  
include a printer lead and  
a tailored dust cover.



### Star LC-20 A popular low cost Mono Printer

Multiple font options from  
front panel  
Excellent paper handling  
facilities  
Simultaneous, continuous  
and single sheet stationery  
240x240 DPI graphics

£149.95

### Star LC 24-10

24 Pin Version of LC10  
Excellent letter quality print  
360x360 DPI graphics  
8 resident fonts  
Additional font cartridges  
available

£214.95

### Star LC200 Colour Printer An excellent value for money Colour Printer

240 DPI 9 pin colour  
Front panel pitch selection  
Push/pull tractor feeds  
Reverse paper feeds  
16K buffer  
8 resident fonts  
Supplied with colour and  
mono ribbons  
Front panel programmable  
Paper park with auto  
single sheet loading

£204.95

### Star LC24-200 Mono Printer

80 column dot matrix  
printer  
7K buffer expandable to  
39K  
10 resident fonts  
Paper park with auto  
single sheet loading  
Supplied with mono ribbon  
Faster than LC24-10  
Extra font cartridges  
available  
Front panel pitch selection  
Programmable from front  
panel

£249.95

### Star LC24-200 Colour Printer

360 DPI 24 pin colour  
printer  
30K buffer expandable to  
62K  
10 resident fonts  
Front panel pitch selection  
Reverse paper feed  
Paper park with auto  
single sheet loading

Supplied with colour and  
mono ribbons  
Front panel programmable  
Extra font cartridges  
available

### Citizen 120D+

A very popular low cost  
mono printer centronics/  
RS232/CBM 64 serial.  
Please specify interface  
when ordering (two year  
warranty)

### Citizen 124D 24 pin Mono Printer

8K buffer expandable  
Push and pull tractor feeds  
3 fonts with excellent  
letter quality  
(two year warranty)

### Citizen Swift 9 pin Colour Printer

8 K buffer  
240x240 DPI colour  
graphics  
Push and pull tractor feeds  
Excellent print quality  
(two year warranty)

### Citizen Swift 24 pin Colour Printer

European Printer of the  
Year 1990  
24 pin version of the 9 pin  
colour printer  
360x360 DPI colour  
graphics  
Push and pull tractor feed

## PERIPHERALS

We stock a large  
range of peripherals, all of  
the highest quality. If you  
can't see the one you  
require please give us a  
call to see if we stock it.



### Mice and Trackballs

**Naksha Upgrade Mouse**  
Amiga/Atari switchable  
Simply the best  
280 DPI resolution  
Ergonomic design  
Smooth, precise operation  
2 years guarantee  
Free mouse mat  
Free mouse pocket  
Now includes  
Operation Stealth from US  
Gold. Only

GOVERNMENT, TRADE AND EDUCATIONAL ORDERS WELCOME





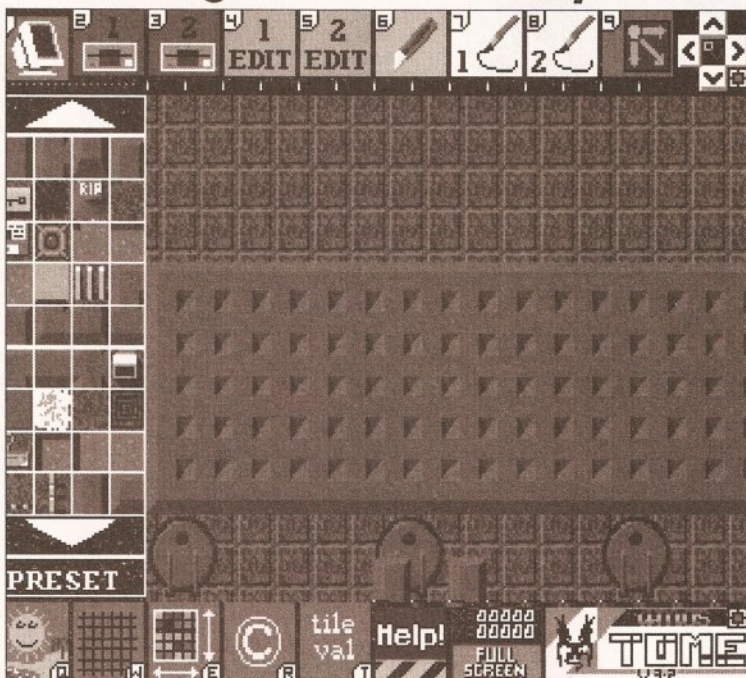


# Amos Action



"Welcome to the AMOS column, the place where every month you can find tutorials and lots of hints and tips for use with Mandarin Software's AMOS BASIC interpreter."  
Phil South

**This month AMOS chief Phil South looks into AMOS TOME, the Oasis NCOMMAND extension disk, plus all the regular hints and tips**



With TOME you save time and memory when creating mapped games. The program works like a collage; simply paste together your building blocks...

## AMOS TOME

TOME is something that's much heard of and rarely seen in AMOS circles, and this is mainly because it's a super extension to the AMOS system which enables you to create mapped games with ease. In fact anything which would require screens to be linked in a certain order can be more easily carried out with TOME installed.

TOME began life as a small utility program on the Atari ST (spit!) which enabled the user to produce game backgrounds which were much larger than the actual physical screen.

Over the last few years the product has been improved, until, when AMOS was finished, an Amiga version of TOME was created. AMOS Total Map Editor is the ultimate MAP designer for use not just with AMOS but with any language on any computer!

The program takes the form of the TOME Editor, where all your bits of map are patched together and saved as a file, and the TOME extension, which is added to the AMOS system to do various map

type commands for you.

The version I got to review is the newest AMOS 1.3 or Compiler compatible version called TOME 3.2, which has recently been supplied free to all previously registered AMOS TOME users, and is included in all new AMOS TOME packages. (Note: If you're still using AMOS 1.23, then I would advise you to upgrade to AMOS 1.32 right now!)

## WHY USE TOME?

The reason for using TOME is that it enables you to create giant

screens in memory using a series of simple building blocks, these areas of memory are called MAPs - for obvious reasons.

You can design a giant MAP the height of many screens and it will not take up nearly as much memory as drawing a picture that big.

TOME works a little bit like a jigsaw or collage in that you first create building blocks called Tiles, you then use the editor to paste these building blocks on to your MAP area thus creating a giant picture.

Tiles are simply square blocks of a picture, cut out using the utility supplied with TOME. This utility cuts up a picture into 16\*16 or 32\*32 pixel blocks and stores them as AMOS Icons in an AMOS Icon bank.

Tiles are created by drawing them on an IFF screen, starting from the Top Left (Tile number 0) and working right, then going to the next line, working right and so on. You then load up the TileMaker program, which converts the picture into AMOS Icons. AMOS Icons are areas of a screen which can be cut out and saved inside a safe area of memory ready for later use. They can be thought of as less flexible versions of AMOS BOBs.

The practical use of TOME can be seen in many different commercial games which have large scrolling areas, *Ghouls 'N' Ghosts* by US GOLD and *Rainbow Islands* by OCEAN/GRAFTGOLD are two good recent examples.

## INSTALLING TOME

Before you can use the TOME commands, or any of the programs on the TOME disk, you need to install the TOME extension on

continued on page 112

National Code 0373 \_\_\_\_\_ Districts with that number

File No	Districts
979	Chapmanslade
980	Faulkland(Somerset)
981	Frome
982	Mells
983	Nunney
984	Westbury(Wilts)

Press any key to return to Workbench, Press ESC key

If you need to locate calls you've made, when all you have is a number, Natcodes is what you need. See Amos Hints and Tips for more details

# AMOS HINTS AND TIPS • AMOS HINTS AND TIPS

Every month I will be printing AMOS hints and programs from my own sources and from you the readers. If you have any hints and tips you want to send me (preferably accompanied by your listing on a disk), send them to: Phil South, AMOS Action, Amiga Shopper, 30 Monmouth Street, Bath, Avon, BA1 2BW. Or you can e-mail me on:

CIX - snouty@cix.compulink.co.uk

Telecom Gold - 74:MIK2077

The Direct Connection - uad1135@dircon.UUCP

Tony Ashton has come up with the ultimate solution to sprite positioning. I'll let Tony tell the story:

"I often find that I need a bob or sprite to appear at a specific location. Also since I sometimes need to flash or change colours, I need to know which colour is at which location in my palette. For this reason I wrote the enclosed tool, which loads up the background scene, and picks up your bob/sprite so that you can position it exactly on the screen while showing the hardware co-ords. Click the sprite in place, and it gives you the screen co-ords. Now you can use the two menus to select the screen and sprite palettes. Highlight the colour you want to alter and the program prints the hex number of the colour to the screen. Now you have a record of all you need. To change the sprite you position, simply alter the sprite number in the XMouse/YMouse Do Loop."

A neat and quick programming util, which you could program as an accessory. What about writing a way of saving the data in a meaningful form to a file or the printer? Any ideas how that could be done? Thanks anyway to Tony. This is an excellent example of the programs I like to see: short, neatly and logically written, and useful too.

## X Sprite - Y Sprite by Tony Ashton

```
' Converts Sprite Co-ords to Screen ↵
Co-ords
' Shows Sprite Register
' Position sprite - click mouse - ↵
shows screen co-ords
' Menu shows colour chart & Hex no.↵
(excl. colour 0 (black))
' Select Hex no. - click left mouse ↵
- Hex no. printed to screen
' Move colour charts to convenient ↵
position by "picking up" with left
' mouse.
' Change to required sprite in X ↵
Mouse,Y Mouse loop
'
Cls
Erase 1
F$=Fsel$("DF0:", "", " Load Background ↵
Picture ")
If F$="" Then Edit
Load Iff F$,0
'
SPR$=Fsel$("*Abk", "", " Load Related ↵
Sprite Bank ")
If SPR$="" Then Edit
Load SPR$,0
Get Sprite Palette
```

```
Curs Off : Paper 22 : Pen 15 : Rem - ↵
or whatever !
'
```

```
Locate 0,0 : Print " X Hard is ";
Locate 0,1 : Print " Y Hard is ";
'
Do
Sprite 9,X Mouse,Y Mouse,1
Locate 20,0 : Print X Sprite(9);" ";
Locate 20,1 : Print Y Sprite(9);" ";
If Mouse Key=1 Then Goto LABEL
Loop
'
```

```
LABEL:
Locate 28,4 : Print "Sprite";
Locate 28,5 : Print "Register";
For C=16 To 31
Locate 28,C-9
Print Hex$(Colour(C),3)
Next C
'
Locate 0,4 : Print "Screen";
Locate 0,5 : Print "Register";
Locate 0,7
For C=1 To 15
Print Hex$(Colour(C),3)
Next C
'
Locate 25,0 : Print X Screen(1);" ";
Locate 25,1 : Print Y Screen(1);" ";
'
```

```
Menu$(1)=" 0 to 15 "
Menu$(2)=" 16 to 31 "
'
For D=1 To 15
Menu$(1,D)=" (IN 1,"+Str$(D)-""+" )↵
(BA 30,12)"
Menu$(1,D,1)=Hex$(Colour(D),3)
Next D
For C=16 To 31
Menu$(2,C)=" (IN 1,"+Str$(C)-""+" )↵
(BA 30,12)"
Menu$(2,C,1)=Hex$(Colour(C),3)
Next C
Menu On
Do
If Choice
Locate 28,25 : Pen 24 : Inverse On : ↵
Print Hex$(Colour(Choice(2)),3)
End If
Loop
```

Tony's excellent program gave me an idea, and I whipped up this little program to show up mouse co-ords. I know the thing which flummoxed me about limiting and reading the mouse over certain points on the screen was positioning. And this program enables you to do this very accurately. Load an IFF picture, preferably the one you want to locate the mouse over, and then just read the figures (and write them down) as you move the mouse over the screen.

## Mouse locator

'A program to show the location of ↵
the mouse on screen, for positioning ↵

over an IFF file of your choice...

```
'
Paper 0 : Clw
Curs Off
F$=Fsel$("*IFF", "", " Load Background ↵
IFF Pic ")
If F$="" Then Edit
Load Iff F$,0
Locate 0,0
Print "Snout's Mouse Co-ord Engine"
Locate 0,1 : Print "X= " : Print ↵
"Y= "
Locate 0,20 : Print "CTRL-C to stop"
Do
Y1=Y Mouse
X1=X Mouse
Y1$=Str$(Y1) : X1$=Str$(X1)
Locate 6-Len(X1$),1 : Print X1 :↵
Locate 6-Len(Y1$),2 : Print Y1
Loop
```

And finally from Graham Jones, a replacement for the Sort routine on page 59 of the AMOS manual. Graham invented this routine as part of an excellent program he wrote called *Natcodes*, which will soon be appearing as licenseware. This program takes the STD codes for different parts of the country and tells you where that place is. The phone book tells you the place and then the number. This way you can type in a code you've dialled and find out where it was you were calling.

## 'Stringsort' by GH Jones

```
'
Screen Open 0,640,256,4,Hires
'
Paper 0 : Clws
Locate 0,0
Print "This routine replaces that ↵
given in the "
Print "AMOS manual for the Sort ↵
command."
Print "It can also be modified to ↵
use variables."
Print
Print "<Press a key>" : Wait Key : Cls
'
Input "Enter max No. of Array values ↵
";N
Print : Inc N
Dim A$(N)
P=1
Repeat
Input "Enter string ↵
(Null to stop) ";A$(P)
Inc P
Until A$(P-1)="" or P=N
Sort A$(0)
For I=N-(P-2) To N
C=I-(N-P+1)
A$(C)=A$(I)
A$(I)=""
Next I
Cls
Print "Sorted array in ascending ↵
order." : Print
For J=1 To N
Print A$(J)
Next J
Wait Key
```

continued from page 110

to your copy of AMOS. Once the extension is installed into AMOS, you can use the AMOS *TOME* commands within your programs or even in direct mode, as they become part of the AMOS Language. Installing *TOME* 3.2 is a little more taxing than previous versions as you have to bolt on the thing manually rather than just running a neat install program. This is no real problem though.

To install the *TOME* tools, just load the *TOME\_INSTALL.AMOS* program into your AMOS 1.3 and run it. Then all you need to do is to select which version of AMOS it is that you wish to install the extension on to. The program will then look for a disk called AMOS: (if you have ASSIGNED this on your hard drive it will also work) and save a program called *TOME.Lib* into your *AMOS\_SYSTEM* directory. Once this is done, you can quit the *TOME\_INSTALL* program.

The next step is to load in the AMOS configuration program. In 1.3 of AMOS it's called *CONFIG1\_3.AMOS*. Next you will have to load the default configuration.

After carrying out the above steps you have to select the *LOADED EXTENSIONS* item. This will give you a listing of all the extensions that have been installed into AMOS. You will probably have something like the following:

- 1: *AMOS\_SYSTEM/Music.Lib*
- 2: *AMOS\_SYSTEM/Compact.Lib*
- 3: *AMOS\_SYSTEM/Requester.Lib*
- 4:
- 5: *AMOS\_SYSTEM/Compiler.Lib*
- 6: *AMOS\_SYSTEM/Serial.Lib*
- 7:
- 8: *AMOS\_SYSTEM/CTEXT.Lib*
- 9: *AMOS\_SYSTEM/Range.Lib*

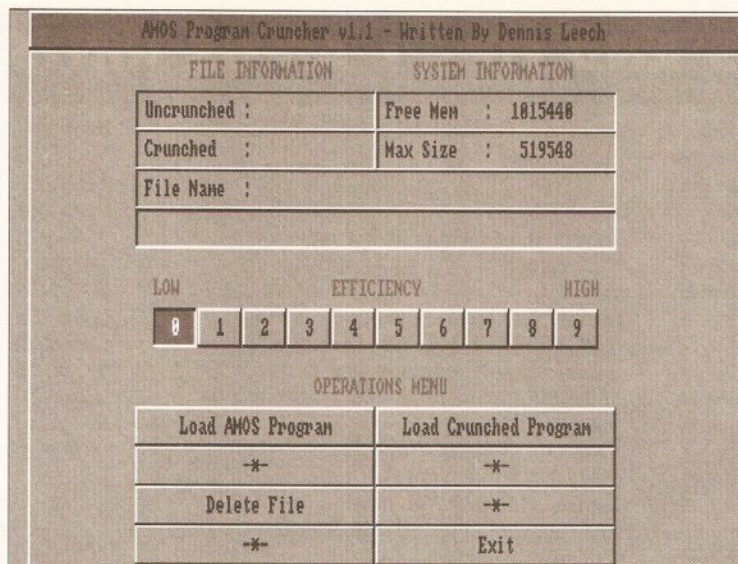
next you have to click on the blank Line 7 and enter:

*AMOS\_SYSTEM/TOME.Lib*

then save that configuration as the default. Now when you reboot your AMOS, it should load up with the *TOME* extension installed, ie it will show up on the list of loaded extensions when AMOS starts.

## DOCUMENTATION

The docs to the *TOME* program are featured on the *TOME* disk as a home built hypertext program. You can click on various text screens and get info about the program, roaming around the text for the manual in an interactive way. I must admit this is quite impressive, but I still prefer paper manuals. All the same the information in the *TOME* hypertext manual was easy to obtain and fast



**NCOMMAND in action.** It adds all the usual characteristics of Workbench 2. Check boxes, number requesters and text lines all feature

too. Not bad for an AMOS program, eh?

## THE TOME EDITOR

The *TOME* editor enables you to create, edit and save all of your game Maps. It is a powerful map editing system which includes simple drawing facilities similar to those found in *Deluxe Paint*, therefore, to get the most out of the system, you must have at least 1Mb of memory.

When you load the editor you see that the screen is divided into four main parts. The first two are icon bars which run along the top and bottom of the screen and control all of *TOME* editor's functions. Each bar is divided into a number of small icons, and as you click on an icon on the top bar a new selection of options appear on the bottom bar.

The next part of the screen at the far left is the tile selector and contains copies of all the Tiles you created with a paint program and *TileMaker*. The scroll arrows at the top and bottom of this area allow you to move through a selection of your Tiles. You can also click on the right hand button while positioned over the map to select a tile. At the bottom of this part of the screen is an icon with the word *PRESET* on it. If you click on this you will see the current range of *TILES* replaced with another set, a user defined set.

The last bit of the screen is of course the Map itself. This is a large area just off centre and to the right of the screen. To move around the MAP you can use the joystick, cursor keys or the little arrows in the far right corner of the top icon bar.

## NICENESS...

Isn't this an Edna Everage song? Sadly no, it's the configure *TOME*

editor menu. The Niceness menu option enables you to configure the editor to suit your own tastes.

The different options displayed on the menu enable you to change the screen from PAL to NTSC, switch on the cursor co-ordinates, adjust the Tile/Icon bar palettes, save the settings exactly how you want them and even return everything back to their default states, the usual config fare.

The palette controls are the

your mouse movements and will create instant mazes as you draw. This is a really good feature, but Auto Map is even better. Whereas Maze Mode will follow your mouse movements to produce a maze of your own design, this mode goes a step further and does all the work for you. This mode can take a while especially when used on large Maps.

## THE TOME COMMANDS

The actual *TOME* extension only works from within AMOS and on files created with the *TOME* Editor. Here are some of the commands you can use from within AMOS:

*MAP DO x,y*

Redraws the map starting from map co-ordinates "x,y" to the current screen. It's suggested by the manual that you draw your maps to a single buffered screen and *SCREEN COPY* them to a double buffered screen, as drawing icons to a double buffered screen is very slow.

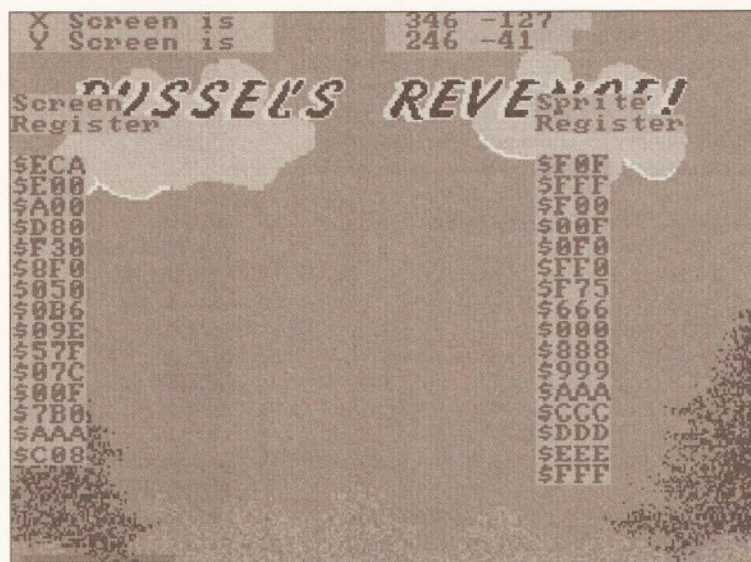
*MAP TOP x,y*

*MAP BOTTOM x,y*

*MAP LEFT x,y*

*MAP RIGHT x,y*

These four commands are exactly the same as the *MAP DO* command,



**The Niceness menu of TOME enables you to adapt the editor to your own taste. The palette controls are similar to those of SpriteX (seen above)**

same as in the AMOS Sprite editor (and *SpriteX*), and enable you to set either the colours of the control icons or the tiles (to your own preference). Simply click on the colours you want to change, and use the sliders to set the RGB values.

## AUTO MODES

There are a couple of automatic modes you should know about. First is Maze Mode. By using this and selecting Tiles for corners and junctions etc, *TOME* will follow

except that they only redraw one edge of the display area. This is very handy for fast scrolling.

*MAP VIEW x1,y1 to x2,y2*

This command creates a window to which *TOME* will limit all of the MAP drawing functions. Obviously the window cannot be any bigger than the currently defined screen.

*TILE SIZE x,y*

This sets the size of the tiles to be used. Normally X & Y will be either

16 or 32, but smaller sizes can be used.

MAP PLOT t,x,y

Places tile "t" at co-ordinates x,y in the map. The same as using the DRAW function within the TOME Editor.

tx=MAP X

Returns the width of the current map in TILES. The width is passed back into variable "tx".

ty=MAP Y

Returns the height of the current map in TILES. The height is passed back into variable "ty".

xt=XTILE(x)

Changes pixel co-ordinate "x" into a TILE co-ordinate. The co-ordinate is passed back into variable "xt".

yt=YTILE(y)

Changes pixel co-ordinate "y" on to a TILE co-ordinate. The co-ordinate is passed back into variable "yt".

tn=MAP  
TILE(x,y)

Returns the tile number as found at MAP co-ordinates "x,y" into variable "tn".

=TILE VAL(x,y,l)

Returns the value from list "l" of the tile at MAP co-ordinates "x,y".

MAP BANK b

Changes the bank used for map storage to bank b.

Normally, the default is bank 6, which you should reserve to the length of the map, and then load the map in. e.g

```
Rem this finds the length of the map
Open in 1, "Mymap.map"
L=Lof(1)
Close 1
Rem This loads the map
Reserve as work 6,L
Bload "Mymap.map",start(6)
TILE BANK b
```

Changes the TILE value list bank from the default of 8 to "b". This is very handy if you want to keep multiple TILE value banks in memory at one time.

=MAP CHECK

This function will run through the map data, checking all the tiles used in the map against those available in the tile bank. If a tile is used in the map which isn't available in the tile bank, then the function will change its tile number to zero. MAP CHECK returns the number of tiles that it has had to change.

## OH GOODIE!

The AMOS TOME Goodies Disk is only available to registered AMOS TOME Users, and includes three new games written with AMOS TOME.

today, and create game maps as big as your ideas.

## NCOMMAND

NCOMMAND caused a bit of a surprise, as it was sent directly to me without any warning whatsoever. The idea behind it is apparently to add some commands to AMOS which make it look just like Workbench 2.0, although why you'd want to do this, I'm not too sure. But it sounded interesting so I thought I'd take a look at it. Initially I got it wrong; the purpose of the program is actually to "emulate a Workbench 2.0 environment" to produce "professional quality displays" with the minimum of fuss and bother.

The program has got a few restrictions; it uses zones 1-210 so you can't use them, it also uses icons 100-115 so you can't use them either. This shouldn't be too much of a problem if you only intend to program professional displays or utils, and besides this subsystem doesn't lend itself to games anyhow.

The NCOMMAND commands are actually a set of AMOS procs, so you don't have to install them directly to the system and they don't become part of the system. You simply load up the NCOMMAND program, add

your program to the bit in the code where it says *add your program here*, and then use the commands in the setting up of your screens.

The command set includes all the usual Workbench 2.0 stuff like buttons, check boxes, rotating requesters, boxes, text lines where you can add text, and number requesters too.

In addition to the above the colour palette has been chosen to coincide with the Workbench 2.0, colours, and the effect, although not exactly like Workbench 2.0 is very professional.

NCOMMAND adds a lot of new instructions (such as Procs) to your programs. Unfortunately this does mean that every program you merge with NCOMMAND goes up in size by the size of NCOMMAND,

which amounts to about 63K. Still, if it's a small util you're creating, you're not going to worry about that too much. The documentation details the new commands at your disposal in a nicely DTPed paper manual, which also has instructions on how to use the program.

## DON'T FORGET...

The AMOS conference which can be found on CIX, has bundles of programs and bags of hints and tips for all AMOS users.

Join CIX by dialling their voice line and asking for details on 081 390 8446, or just log on to 081 390 1244 and have your credit card ready.

## TO SUM IT ALL UP

NCOMMAND is well programmed. Although it is not an exact replica of Workbench 2.0 buttons etc, it's good enough to pass for it at a glance. And this is a very clever way of adding commands to the system without actually building an extension module, as Aaron Fothergill did for TOME.

No sir, this is straight AMOS, and very good it is too. Congratulations are in order to the author Dennis Leech for coming up with such an original and well organised utility.

## TIME TO GO

OK, that's all we have time for this month. Sad to say there was no room for our usual AMAL tutorial or AMOS PD reviews, but no big loss, we'll catch them again next time. Join me again next issue for more AMOS action. See you then! **AS**

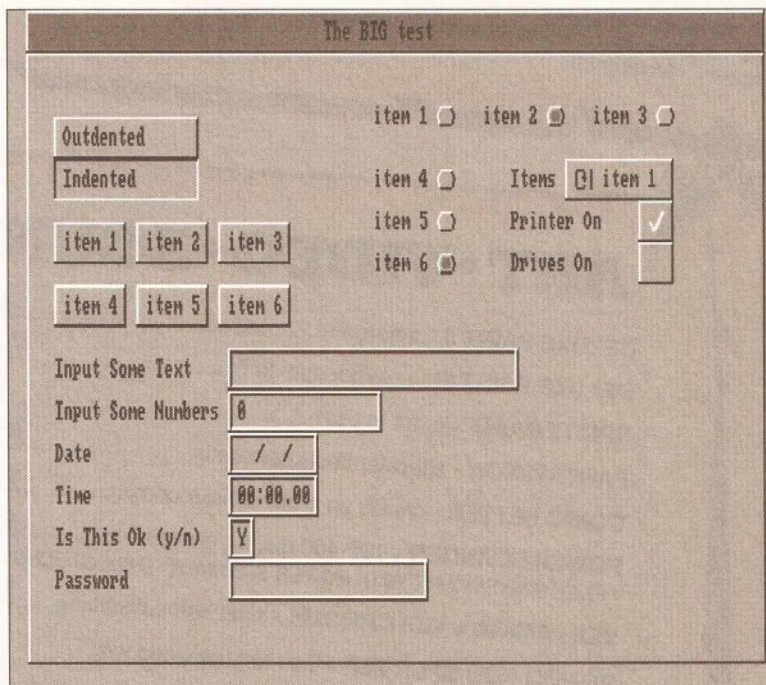
## SHOPPING LIST

AMOS can be obtained from your local stockist or from:

Europress Software  
Europa House  
Adlington Park  
Macclesfield  
SK10 4NP  
☎ 0625 859333

All AMOS PD software and licenseware can be obtained from:

Deja Vu Software  
(aka: the AMOS PD Library)  
25 Park Road  
Wigan  
WN6 7AA  
☎ 0942 495251



If you're thinking that this looks familiar, then you'd be right. NCOMMAND's purpose is to emulate that authentic Workbench 2 look and feel

These are: *Magic Forest II*, a horizontal scrolling platform game. *Green Flag*, an isometric 3D scrolling game. *The Dungeon*, a dungeon Master style 3D dungeon game. All three are supplied in AMOS 1.3 form, so that you can use the routines in your own games. The TOME Goodies Disk 1 costs £5 and is only available to registered TOME users who have sent in their registration card.

## CONCLUSION

AMOS TOME is just one of those things you *have* to have if you're serious about AMOS. TOME costs just £24.99 (or £19.99 to AMOS Club members) from Shadow Software. The next upgrade is said to be TOME 4, release date expected to be some time around March 92. But don't wait for that. Install TOME

# DYNAMITE



## AMIGA A500 PLUS PACKS

### STANDARD

Latest enhanced Amiga from Commodore with 1MB of RAM, Kickstart and Workbench Version 2.04. Complete with mouse, TV modulator, manuals etc.

**£329.99**

### CARTOON CLASSICS

Contents as standard pack plus the following:- The Simpsons • Captain Planet • Lemmings • DeLuxe Paint III

**£349.99**

### DYNAMITE HEROES PACK

Latest enhanced Amiga from Commodore with 1MB of RAM, Contents as Cartoon Classics Pack plus:- Robocop • Batman-The Movie • Indiana Jones • Ghostbusters II • Microswitch Joystick • Dustcover • Mousemat

**£369.99**

## COLOUR MONITORS

### DYNAMITE EXCLUSIVE

#### PHILLIPS 3332 15" FST TV/MONITOR

High quality FST TV with dedicated monitor output. Complete with remote control, sleep timer, cable and a 2 year guarantee!

**PLUS FREE F19 STEALTH FIGHTER**

All this for the DYNAMITE price of

**£239.99**

#### PHILLIPS 8833 MKII DEDICATED STEREO MONITOR

Complete with cable

**£229.99**

#### COMMODORE 1084S STEREO MONITOR

Complete with cable

**£229.99**

## ART • VIDEO • SOUND

DELUXE PAINT II - a bargain! ..... **£9.99**

DELUXE PAINT III - upgradeable to IV ..... **£19.99**

SPRITZ PAINT - joyful to use! ..... **£9.99**

FANTAVISION - special effects generator ..... **£14.99**

COMIC SETTER - create your own colour comic ..... **£19.99**

POWER SCANNER - 100-400 dpi, 64 greyscales ..... **£99.95**

\* FREE SPRITZ PAINT WITH POWER SCANNER WHILE STOCKS LAST!

VIDI - AMIGA + VIDI CHROME - start frame grabbing ..... **£79.99**

ROMBO RGB SPLITTER - Can also be used with DIGIVIEW ..... **£59.99**

COMPLETE COLOUR SOLUTION - everything you need + PHOTON PAINT II ..... **£129.99**

MUSIC X JUNIOR - brilliant for beginners ..... **£19.95**

4 PORT MIDI + MUSIC X JUNIOR - excellent combo ..... **£29.95**

MUSIC X - latest version for the professional ..... **£49.95**

## KLONDIKE DEALS

**DYNAMITE COMPUTERS  
DIG DEEP FOR  
THESE SPECIAL  
BARGAINS....**

### CARTOON CLASSICS SOFTWARE PACK

Bart Simpson .v. Space Mutants • Lemmings • Captain Planet ..... **£19.99**

### 'SCREEN GEMS' SOFTWARE PACK

Back To The Future II • Nightbreed • Shadow Of The Beast II • Days Of Thunder ..... **£19.99**

OFFICIAL COMMODORE A501 1/2 Meg Upgrade ..... **£29.99**

### ASTRA TEN GAME SOFTWARE PACK RRP £279

DataStorm • Dungeon Quest • E-Motion • Grand Monster Slam • Kid Gloves • Powerplay • RVF Honda • Shuffle Pack Café • Soccer • Tower Of Babel ..... **£19.99**

GFA BASIC 3.5 INTERPRETER RRP £49.95 ..... **£14.99**

# COMPUTERS

## THE PRICE BLASTERS

### DISK DRIVES

**FREE DYNAMITE DISK DRIVE STARTER KIT WITH EVERY DRIVE PURCHASED FROM THIS SECTION!**  
 Starter Kit comprises:- 3 1/2" disk drive cleaner kit • Pack of ten 3 1/2" disks • 40 capacity lockable disk box

3 1/2" INTERNAL	
DIRECT A500 replacement drive, no modifications required .....	£49.99
3 1/2" EXTERNAL	
PC880A POWER DRIVE featuring anti-click .....	£49.99
CUMANA CAX 354 slimline drive .....	£54.99
PC880B POWER DRIVE featuring Blitz/Virus hardware .....	£69.99
DUAL PC880B POWER DRIVE with built in power supply .....	£119.99
<b>GVP SERIES II HARD DRIVE</b>	
GVP Series II HD8 + 52Mb hard disk .....	£359.99
GVP Series II HD8 + 105Mb hard disk .....	£539.99

### STAR/CITIZEN PRINTERS

<b>CITIZEN 120D+</b> RRP £233.83 144/30 CPS, 4K Buffer, 2yr guarantee	£114.99
<b>STAR LC20</b> RRP £233.83 150/38 CPS, Multiple Fonts, 240x240 dpi	£129.99
<b>CITIZEN SWIFT 9</b> RRP £280.83 192/48 CPS, colour kit option, 2 yr guarantee	£169.99
<b>STAR LC200 COLOUR</b> RRP £304.38 185/40 cps, with colour & mono ribbon	£184.99
<b>CITIZEN 224 (New)</b> RRP £316.88 192/80 cps, colour kit option, 2 yr guarantee	£214.99
<b>STAR LC24-200 MONO</b> RRP £351.33 220/55 cps, 24 pin 360 dpi, 7K Buffer expandable to 39K	£219.99
<b>CITIZEN PN48 (New)</b> RRP £381.88 Portable A4 notebook printer with 'laser quality'	£249.99
<b>STAR LC24-200 COLOUR</b> RRP £410.88 220/55 cps, 24 pin colour, 30K buffer	£259.99
<b>CITIZEN SWIFT 24e</b> RRP £428.88 216/80 cps, colour option, 2 yr guarantee	£269.99

### ACCESSORIES

Parallel Printer Cable.....	£3.99
Universal Printer Stand.....	£3.99
500 sheets of continuous paper.....	£4.99
Colour kit option for Swift 9/224/24e.....	£26.95

### ACCESSORIES & EXTRAS

Pack of 10 blank 3 1/2" DS/DD disks ....	£4.99
Box of 50 blank 3 1/2" DS/DD disks ..	£19.99
512K RAM expansion (Plus compatible) .....	£19.99
512K RAM expansion with clock .....	£24.99
1Mb RAM expansion (upgrades Plus to 2Mb) .....	£49.99
Amiga dustcover + Resolution 101 game .....	£4.99
Euromouse (quality replacement mouse) .....	£12.99
40 capacity lockable disk box .....	£3.99
80 capacity lockable disk box .....	£5.99
150 capacity 'stronghold' box (lockable) .....	£17.99
6mm quality mousepad .....	£2.99

ALL PRICES INCLUDE VAT at 17.5%

### HOW TO ORDER



Call us NOW on  
0234 214212



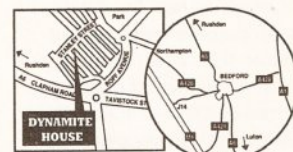
Send cheque or postal order to

DYNAMITE COMPUTERS  
Dynamite House  
44a Stanley Street  
Bedford MK41 7RW

NB Please write cheque guarantee card number on the reverse of cheque to ensure same day clearance (cheques without card number subject to 10 day clearance)

Call into our showroom for a fast and friendly service

Retail/Queries 0234 364428



### DELIVERY CHARGES\*



£2.95 for orders less than £100

£4.95 for orders over £100

Next day express service £7.50

\*Charges applicable to UK mainland only

### PLATINUM SERVICE CARD

Only £19.99!

Features: •Next day courier delivery

•30 day exchange for new

•1 year guarantee giving free collection and delivery should a fault occur

DYNAMITE

TO PLACE YOUR ORDER PHONE THE DYNAMITE HOTLINE

# 0234 214212

DYNAMITE HOUSE, 44a STANLEY STREET, BEDFORD





"If MIDI throws you into a muddle, then that's where I step in. Magic with your MIDI was never easier."

Paul Overaa

**T**he other day I was asked why the MIDI Standard doesn't get altered to include more channels. Although a lot has been written about MIDI messages, their layouts and their uses, no-one seems to have dealt with the structure of these messages from an 'overall plan' type of viewpoint. As it is the MIDI spec's status byte structure definition which limits the MIDI messages that can be formed it seemed worthwhile spending some time discussing how the various message classes were created.

Before this, a few words about the main categories of messages supported by MIDI: At the highest level MIDI distinguishes between Channel messages (those which contain a channel number) and System messages. Channel messages can be one of two types – Voice messages, which are concerned with creating melodies/rhythms and selecting various sounds etc, and Mode messages which define different message interpretation schemes. System messages deal with the transmission of information of interest to the MIDI system and they come in three forms: Common, Real-Time and System Exclusive (SYSEX).

## STATUS BYTE LAYOUT

For the moment it's the status byte layout which concerns us: Status byte values have the most significant bit of the byte (bit 7) set high. However this distinction is only the beginning of the hierarchy within the multi-byte MIDI message system, but it's not hard to see that this scheme immediately limits the number of different messages that MIDI can use there are effectively only 7 bits of the status byte available to represent different message types).

Now let's look at the status byte bits b6 – b4. These can form eight different patterns: 000, 001, 010, 011, 100, 101, 110 and 111. Seven of these are used to define channel

# Don't skip

*In this month's instalment Paul Overaa takes a look at the MIDI standard and explains just how the various classes of MIDI messages can be categorised...*

Figure 1: Channel messages in the status byte

b6	b5	b4	Message Type
0	0	0	Note Off
0	0	1	Note On
0	1	0	Polyphonic key pressure
0	1	1	Control Change
1	0	0	Program Change
1	0	1	Channel Pressure
1	1	0	Pitch Bend Change

messages as shown in Figure 1.

These bit patterns have been designated as channel messages. Where does the channel number go? It's stored in the lowest four bits b3-b0 of the status byte number. Channel numbers 1-16 are stored internally as the numbers 0-15, so a note-off status byte for channel 1 is represented as shown in Figure 2.

discussion as it has nothing to do with channel messages at all – it has been reserved for various system messages as shown in Figure 3.

For these System messages, bits b6-b4 can only take the value 111, because all of the other combinations have been used to represent channel message types. So how are the System messages

that, with the exception of SYSEX status byte (which defines its own special group of 'non-standard' messages), are collectively called System Common messages. Figure 4 shows how the bit values fit into the overall scheme of things.

If bit 3 is set high (ie to 1) we get the sub-group, called the real-time message group as shown in Figure 5.

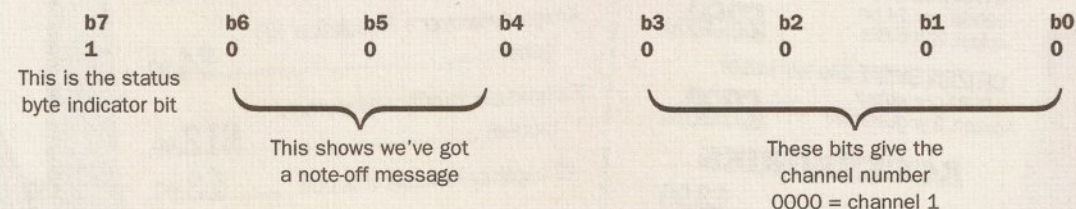
That's covered the ideas behind the status byte, now we can look at the purposes of the messages...

## SYSTEM MESSAGES

The messages vary in length... some are single byte, some contain 2-3 bytes, others more. Real-time (RT) messages contain a single byte, ie the status byte is actually the whole message. There are RT messages for Start (decimal 250), Stop (decimal 252), Continue (decimal 251), Active-Sensing (decimal 254), System-Reset (decimal 255), and Timing-Clock (decimal 248).

Start, Stop and Continue announce the obvious... if for

Figure 2: A note-off message to channel one



This, incidentally, explains why the 16 conventional MIDI channels limit exists: Four bits b3-b0 can only store 16 different patterns... 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, and 1111. So, unless the whole of the status byte was redefined and expanded, it's impossible to include more than sixteen channels.

In defining the channel message status bytes, the bit patterns have taken up all of the combinations in the upper four bits except for one... the pattern 1111. I didn't include this combination in the above

represented? It's easy... System messages *do not* have a channel number, so bits b3-b0 can be used to identify 16 different types of system message. What happens is that b3 is used to indicate one of two message sub-classes. If the value of b3 is zero we have a group

instance you hit the start button on a drum machine, it will transmit a 'start' message. Active Sensing is one of those background messages you'll never see in your sequencer event lists – units which implement active sensing can tell whether there are communications problems by

Figure 3: The status byte signifying a system message

b6	b5	b4	Message Type
1	1	1	System Messages

# a beat!

**Figure 4: The various MIDI system messages**

b3	b2	b1	b0	Message Type
0	0	0	0	SYSEX Message
0	0	0	1	Midi Time Code
0	0	1	0	Song Position Pointer
0	0	1	1	Song Select
0	1	0	0	Undefined
0	1	0	1	Undefined
0	1	1	0	Tune Request
0	1	1	1	End of System Exclusive

transmitting 'dummy data', streams of active sensing messages, during the times when there is no other data on the MIDI line. If a unit which was receiving such messages suddenly finds that there is no real MIDI data and no active sensing messages either then it will switch off its sound generators as a safety precaution. When you buy a piece of MIDI equipment you'll get a manual including a MIDI Implementation Chart. This gives a standardized summary of the facilities on offer and you'll see details on active sensing. System Reset is a command which will force a piece of equipment to

246) asks all synthesizers to tune their oscillators. Song Select (decimal 243 followed by a number from 0 to 127) enables you to specify a song by providing a reference number – if, for instance, you've a selection of songs programmed into a drum machine you can use this message to make selections automatically. Song Position Pointer messages allow songs and sequences to be started from places other than the beginning. They're three bytes long and consist of a status byte (decimal 242) followed by the two data bytes which identify the start. For the

**Figure 5: The real-time message group**

b3	b2	b1	b0	Message Type
1	0	1	1	Continue
1	0	0	0	Timing Clock
1	1	0	1	Undefined
1	0	1	0	Start
1	1	1	1	System Reset
1	1	0	0	Stop
1	0	0	1	Undefined
1	1	1	0	Active Sensing

assume a 'just switched on' state.

Timing clocks are transmitted by sequencers, drum machines, master keyboards etc, as a means of keeping everything 'in sync' – since they provide a timing reference to be used by all units, it's obvious that only one MIDI unit should be generating them at any one time. You'll also read about three common messages... Tune Request (decimal

techies the gory details are shown below... the two data bytes are sent as low byte followed by high byte and because only 7 bits of each byte are used, you combine the lower 7 bits of each data byte to produce a 14-bit number and it's this number which identifies the starting position – it identifies the MIDI beat, (1 MIDI beat = 6 clocks) and corresponds to a starting resolution of one semi-

quaver in a song, see Figure 6.

## CHANNEL MESSAGES

Channel messages constitute the largest of the MIDI message groups. In the MIDI spec these messages are divided into two sub-categories: 'Mode messages' affect the way that units interpret the data they handle, and 'Voice messages' are concerned with sound production.

Synthesizers contain sound generators, 'voices'. 'Voice assignment' is the term given to the process of routing note-on and note-off data from the keyboard (or MIDI terminals) to the voice circuitry – so that the right notes are played with the right sounds. With MIDI the

to four possibilities and the MIDI standard sets out the definitions shown in tables 1 and 2...

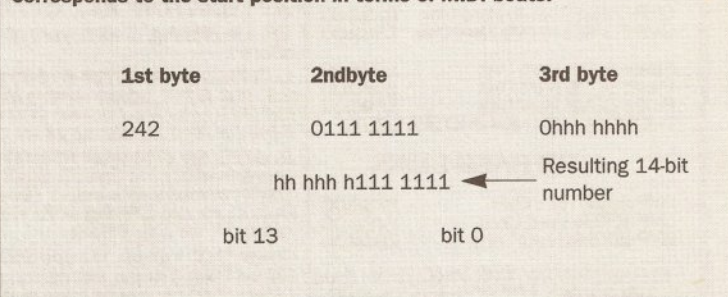
## THOSE MIDI MODES

Well, that covers the details. Now for the question which everyone asks at one time or other... What does it all mean? Here's a condensed version of the main ideas: Mode 1, because it disregards channel info, is only useful as a 'fail safe' mode. A unit set to Mode 1 will respond to any channel message it receives. It's useful for trouble-shooting, checking leads etc, but there are few occasions where you would deliberately want a single unit to respond to data from all channels!

Mode 2 (OMNI ON / MONO) is another mode which ignores channel selectivity. You can forget about this mode – rumour has it that the definition was a mistake and it has no obvious advantages or uses!

Mode 3, (OMNI OFF/POLY ON) is the most widely used of all the MIDI modes. Synthesizers will transmit polyphonic data on the selected

**Figure 6: Song Position pointer message layout... a 14 bit number corresponds to the start position in terms of MIDI beats.**



relationship between the sixteen available channels and the voices also has to be defined and here the MIDI standard specifies separate modes of operation. These modes are usually selectable by front panel operations or by remote selection via an appropriate MIDI message.

There are two variables involved: The first option is whether the unit is going to receive data on all 16 channels or not – the term for the former state is OMNI ON, that for the latter is OMNI OFF. In short, when you select a mode that is OMNI ON based... every channel message that the unit receives will be acted upon, meaning that you lose the benefits of channel selectivity! The other characteristic under your control is whether you specify polyphonic or monophonic operation. Here the situation tends to get complicated... MONO, when on, restricts the assignments of voices to one voice per channel. With POLY ON (POLY ON = MONO OFF), any number of voices can be assigned by the MIDI unit's voice assignment mechanism.

The various pairs of combinations of these choices lead

channel, and any receiving equipment which is in this mode (and set to receive on the same channel) will interpret and use the information to play polyphonically.

Mode 4 is special. With Mode 4 selected each synthesizer voice gets its own channel number. With an 8 note polyphonic synthesizer tuned to basic channel 1 the first voice responds to data received on channel 1, the second to data on channel 2 etc. In other words Mode 4 uses a 'user-definable' set of MIDI channels for its operations. As far as applications (eg MIDI guitar control) go, there are some grey areas but there's no doubt that it does have some interesting potential.

Although there are only four official MIDI modes, the facilities now offered on some expander units have added an interesting new slant to the MIDI mode debate. Many multi-timbral expanders and synthesizers can now be set up to behave as though they are two or more separate synthesizers each controlling their own voices and responding to their own channels.

continued on page 119



MODE	OMNI	POLY	
1	ON	ON	Voice messages are received on <i>all</i> channels and assigned to voices polyphonically.
2	ON	OFF (ie MONO ON)	Voice messages are received on <i>all</i> channels but control only one voice.
3	OFF	ON	Voice messages are received on the selected channel N and are assigned to voices polyphonically.
4	OFF	OFF (ie MONO ON)	Voice messages are received in voices N, up to (N+M-1) and are assigned monophonically to voices 1 thru M. The number of voices is specified by the third byte of the MONO Mode message.

**Table 1: Receiving Equipment Mode Definitions (for a receiver assigned to a Basic Channel N)**

continued from page 117

Some of Mode 4's uses have therefore been rather quenched by the multi-timbral facilities now available with many units.

Explaining the MIDI Mode messages is easy – they are 3 bytes long and have the same status byte (written in binary as 1011nnnn where the lower four bits 'n n n n' are the binary form of the channel number). The data bytes which follow enable you to set Omni Mode On (decimal 125, 0), Omni Mode Off (decimal 124, 0), Mono Mode On (decimal 126 followed by a byte specifying the number of channels) and Poly Mode On (decimal 127, 0). There are a few other messages, including local control messages and a panic button 'All Notes Off' message, which have also been placed in the Channel Mode Message group.

## VOICE MESSAGES

The MIDI channel 'voice' messages are about sound production. There's quite a collection but the ones which get used more than any other are those which turn notes on and off. When you press a key on a synthesizer keyboard three pieces of data get transmitted – first a status byte which announces that a key has been pressed, secondly a number identifying the key itself (Middle C is assigned a value of 60 and for every semitone above or below this the note number changes by plus or minus one). Last is a number which tells the receiving equipment how hard this key has been hit. In MIDI terms the last piece of information is called the velocity byte and at the end of the day (and in less than

1/1000th of a second) this travels down the MIDI lines as in Figure 7.

This message format (with one exception that I'll look at in a moment) is fixed and this means

MODE	OMNI	POLY	
1	ON	ON	Voice messages are
2	ON	OFF (ie MONO ON)	One voice message is sent on Channel N
3	OFF	ON	Voice messages are transmitted on selected channel N
4	OFF	OFF (ie MONO ON)	Voice messages for voices 1 thru M are transmitted in voice channels N thru (N+M-1)

**Table 2: Transmitting Equipment Mode Definitions (for transmitters assigned to channel N)**

course that all MIDI keyboards have to transmit velocity data – even those without touch sensitivity. The big difference is that touch sensitive keyboards work out how hard you've hit the keys, and translate that 'pressure' into a velocity value between 1 and 127. Non touch sensitive keyboards transmit a fixed 'default' value (decimal 64) instead.

## NOTE OFF MESSAGES

As you release keys on the keyboard, streams of messages must be transmitted to indicate the notes to be turned off. The MIDI standard allows two ways of doing this...

Firstly the keyboard can send an equivalent 'Note-Off' message. This contains three bytes but although it

takes the same form as a 'note-on' message, it uses a different status byte as in Figure 8.

A second way that 'Note-Off' information can be transmitted is by sending 'Note-On' messages using zero for the velocity value. With this approach the equipment loses the ability to transmit a 'release velocity' value, but it gains an important potential advantage in another area.

When the second approach is used the messages which turn notes on and off are transmitted in terms of streams of MIDI's 'Note-On' messages. Why is that useful? It just so happens that part of the MIDI specification defines 'running status', which allows all messages after the first one to be transmitted without their status bytes – providing they form part of a stream of identical message types. The benefit of turning notes on and off using Note-On messages is that running status enables a third of the transmitted data to be eliminated – quite a significant saving.

Well, that's about it as far as note-on and note-off messages are concerned. The Program Change message, discussed earlier in the series, is another from the voice message group and there are a few more voice messages that ought to

Because of this, their indiscriminate use can lead to two types of problems... firstly sequencer memory can get 'eaten up' like there's no tomorrow, and secondly you may experience 'MIDI-clogging' problems owing to the fact that too much information is being transmitted.

The Aftertouch message layouts were provided last month so there's no point in repeating the details. Pitchbend (Pitchwheel) messages are sent whenever the pitch wheel or lever changes position and they take the form shown in Figure 9

## CONTROLLER MESSAGES

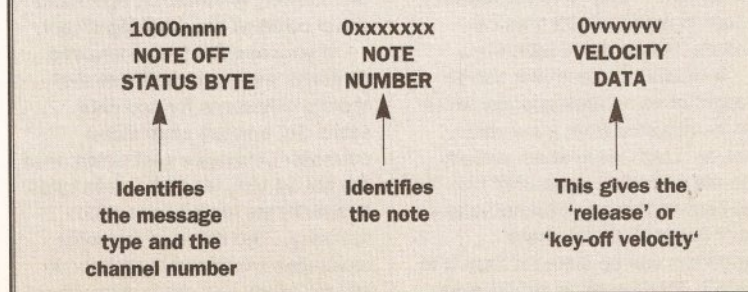
MIDI's controller messages are an area which cause more confusion than expected. Part of this stems from the fact that many people are under the impression that there are many different controller messages involving switches, continuous controllers, and both high and low resolution facilities. It's true that several types of MIDI controller are available – but there is only *one* type of MIDI controller message and once you understand its format and the way it is used, all of the categories of controller use will fall into place.

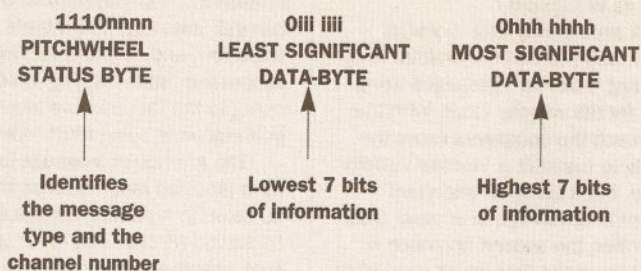
A MIDI controller message requires three bytes and takes the general form shown in Figure 10.

The first byte, the status byte, is the value B0 hex combined with a channel number. The two bytes which follow are both MIDI data bytes, so they can *only* take values between 0 and 127 (0 hex – 7F hex). The second byte of a controller message is termed a controller number and although it could take values as high as 127 it's limited to 120 for controller number purposes – the MIDI specification has used values 121-127 for other purposes (local control, all notes off and the mode messages). Another simplification is that, at the moment, the controller numbers 102-120 are undefined, so you don't need to worry about them. We can also knock out the 96-101 controller range because these have been assigned for rather specialised increment/decrement/parameter-number uses.

That, to all intents and purposes, brings us down to a set of controller numbers from 0 to 95 and these are

**Figure 8: Equivalent layout of a MIDI note-off message**



**Figure 9: General form of a pitchbend message**

split into two groups, switch controls and continuous controls, which I'll now try to explain in detail...

## SWITCH CONTROLLERS

The switch controllers (there are 32 of them) are the easiest to deal with because they're just the MIDI equivalent of an on/off switch. The third byte of a switch message can take one of two values... 0, which means that the switch is off, and 127 (ie 7F hex) which means that the switch is on. Switch messages therefore look like Figure 11.

For controller switch use, the controller data values which fall between 0 and 127 (ie values 1-126) are meaningless and are ignored. This definition is a bit wasteful of byte-space but a more efficient bit encoding scheme, eg perhaps representing 7 switches per single controller message (using one bit per switch), would have meant that such messages would take longer for the receiving equipment to decode.

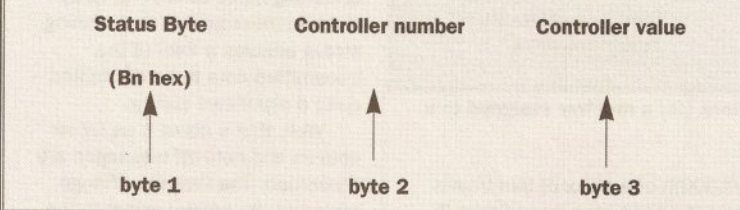
Examples? If you look at the MIDI Implementation sheet for Yamaha's TX81Z expander you'll see that switch numbers 64 and 65 are used to control sustain and portamento. Since 65 decimal = 41 hex it's not too difficult to work out that, if the unit was set to channel 16, the message which would turn the portamento on would consist of the three values BF hex, 41 hex, and 7F hex. The equivalent message which turns the portamento off would just use a zero last byte rather than the 7F hex 'switch on' value.

## ONGOING MESSAGES

Controller numbers 0-63 are known as continuous controllers and these probably are the worst offenders as far as confusion goes. To start with they do *not* represent 64 different 'controllers' - they're effectively only meant to represent 32 physical devices. Here's the reason why...

A continuous controller can be thought of as an analogue dial which can be adjusted from a low value, zero, to a high value which actually has the numerical value 3FFF hex. If you convert these to decimal values you'll find that means these controllers can be adjusted from 0 to 16383. These number ranges were

selected so that a high resolution, (ie a 'dial' containing many different intermediate states) could be used. From the viewpoint of those who defined the MIDI standard the use of continuous controllers presented two problems: first of all the proposed

**Figure 10: General form of a MIDI controller message**

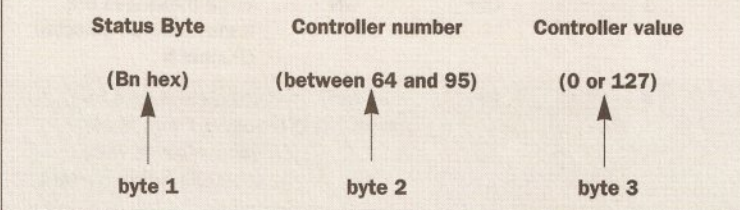
high resolution might not always be needed. Secondly, because a single controller message can only carry a 7-bit controller value, some alternative means of representing the full controller value (which needs 14 bits) had to be found.

The solution, which has been subjected to more than its fair share

is being transmitted? Each continuous controller is assigned two different controller numbers. One number says 'here comes a new value for the highest 7 bits of the controller data value', the other says 'here comes a new value for the lower 7 bits'. Which values are used? Controllers 0 - 31 carry the most significant bits (MSB), Controllers 32-63 carry the least significant bits (LSB). Controllers 0-31 and controllers 32-63 are paired and in reality each pair (0/32, 1/33, 2/34 etc.) represents the values for the same physical controller.

The MIDI standard, in an attempt to save space on the communications lines, allows for partial transmission of these continuous controller data values. It

states that it is only necessary to send both parts of the controller value if... both parts are needed and both parts have changed in value. In other words if a manufacturer only wants to implement some particular low resolution controller facility it can arrange to transmit and recognize controller messages using a

**Figure 11: MIDI switch control messages**

of criticism, was as follows: If you look at the 3FFF hex maximum value and express it in binary form you'll see that it does indeed need 14 bits since 3FFF hex = 11 1111 1111 1111 binary. If you take those bits and divide them into two groups - you have two groups each consisting of 7 bits. The left hand group contains the most significant bits of the number, whereas the right hand group contains the least significant.

If you return to the generalized controller message you'll see that there's only space for *one* data value. So, how are continuous controller messages sent which need the full 14 bits, ie need 2 data bytes, to specify the 'controller position'? It's easy... *two* different controller messages are transmitted. How do you tell which part of the data value

that the unit recognises controller number 1 messages, but doesn't recognise controller 33 messages. It shows that Yamaha has implemented low resolution mod-wheel control.

Nothing can change this, so even if a keyboard could send high resolution mod-wheel messages to the TX81Z (which would consist of a combination of controller 1 and controller 33 messages) the TX81Z, because it only understands the controller 1 messages, would still provide low resolution response.

The use of high resolution controllers means that more data has to be transmitted but, it's not always necessary to transmit both parts of the controller's data values. If some recognised control movement results only in a change in the LSB part of the controller value... then only one controller message (the LSB part) need be sent.

The controllers specified in the MIDI standard are a description, a logical blue-print, of 32 switches and 32 'dial type' continuous controllers. The relationship between the controllers and particular effects is 'soft', ie it exists only in the eyes of the software controlling the transmitting and receiving equipment. There's no reason why controller 1 messages, sent by an instrument which is registering the effects of mod-wheel changes, couldn't be mapped by the receiving equipment to some other synthesizer/voice characteristic. This is the idea behind the use of assignable controllers which let you decide which controller number will be used when a dial or pedal is moved, and lets you decide which effect will be modified when a controller message is received. Many controllers now have standard recommended uses (1=mod-wheel, 2=breath-control, 7=volume etc.) but it's unlikely that we'll ever see 100% standardisation, so synths and expanders with assignable controllers will always give you that bit more flexibility.

Next month I'll complete these 'techie' discussions by looking at MIDI's SYSEX messages, at some fault-finding/diagnostic schemes. I'll also be talking about areas which become important as you get into more advanced MIDI use... **AS**

## JARGON BUSTING • JARGON BUSTING

**Bit positions** - It is common practice to label the 8 bits of a byte (just another name for an eight bit binary number) as b7, b6, b5...b0 with bit b0 being the least significant bit, ie the one on the far right of the number. Bit 7 is therefore the most left hand bit - the one which, when set high, indicates a status byte.

**Portamento** - A sound effect created by rapidly sliding from one note to another - today almost all synthesizers have in-built circuitry for producing portamento effects.

**TX81Z** - A programmable 8 note polyphonic sound expander module by Yamaha.



...ing the complete set? – well get your back issues NOW while you can!



## BACK ISSUES

### Did you miss these Issues...?

Both issue 1 and issue 6 are already sold out, so grab the rest while they're still available. Use the form opposite to claim your back issues of Amiga Shopper for only £1.50 each including post and packing.

### ...This is what you missed:



#### Issue

- 2 Buying a paint package plus reviews of pagestream, Colourpic, Sequencer 1, KCSPowerboard, and Scala.
- 3 Accelerating the Amiga plus reviews of Bars & Pipes, Daatascan, ATonce, DynaCADD and Basic languages.
- 4 Buying a printer plus reviews of Pro-24, Real 3D, Wordworth, TEX, Imagine and Superbase 4.
- 5 Amiga Answers special plus reviews of Spectracolor, Superbase 4, Touch-Up, Action Replay, VideoCenter Plus and more.
- 7 Buying a 3D graphics package, plus setting up a DTP system, learning BASIC and MIDI music. With reviews of Vista Pro and a font designer.
- 8 A hardware and add-on buyer's guide special, tons of tips for Professional Page and an in-depth look at the HAM-E colour system. Plus reviews of Invision Plus and Dr T's KCS.
- 9 The definitive guide to the A500 Plus, down-loading satellite images, choosing sequencers, using video and hands on advice for ProPage 2.
- 10 A special graphics issue featuring the Amiga's top art packages plus reviews of Audition 4, Sound Enhancer and a guide to desktop directing with Showmaker.



## So, don't miss any more!

Why don't you subscribe now to ensure your copy each month? For just £14.95 you will get the next 12 issues delivered direct to your home plus a special sampler disk – a gift with our compliments.

Such has been the demand for our demo disks that we've now decided to offer you all three of these fabulous floppies for just £1.75! Yes, that's £1.75. Just use the order form opposite and they'll soon be on their way.

**DISK 1** Contains demos of AMOS from Mandarin – the programming language that took the Amiga world by storm, Devpac 2 from HiSoft – the assembler that's used by the professionals, Real Things from RGB Studios – it brings things to life in Deluxe Paint III and lets you design, build and fly your own butterfly.

**DISK 2 WORDSWORTH** This brand new word processor from Digita International is described by the company as 'a writer's dream' and you'll be able to see if it is for yourself with this demo. You will also get a voucher enabling you to buy Wordworth for just £99.95 – that's a saving of £30 on its normal price! **SEQUENCER ONE** In issue 2 of Amiga Shopper we said of this new sequencing

program from Gajits Music Software "it has to be the budget priced sequencing software for the Amiga." Why not find out for yourself and take advantage of our special voucher offer to obtain a free disk of exciting new samples for use with Sequencer One?

**3D CONSTRUCTION KIT** If you've ever wanted to create your own three dimensional worlds then now is your chance with Domark/Incentive's brand new game creation program. It allows you to use Incentive's famous Freescape programming system to create your own stunning three dimensional games – without a scrap of programming knowledge. Don't be fooled though, this is a utility that will put great programming

power at your fingertips.

**DISK 3 SPECTRACOLOR:** The new entry into the HAM painting arena looks set to upset the appletart. Try its 4,096 colours and animation facilities out for yourself with the demo version. You can wrap brushes on to three dimensional objects, and try out the rub-through effect for blending two images.

**SUPERBASE PROFESSIONAL 4:** You have read a lot about the Amiga's premier database, and now here's your chance to try it out for yourself. Superbase is a fully relational system with its own programming language, and with it you can set up a database for any purpose you like. It can even handle graphics, as well as numbers and text.

...but better still, book up for the year ahead – and never miss out again



Keep your  
copies of  
Amiga  
Shopper safe  
and protected  
right from the  
very start!

*only*  
**£4.95**  
**EACH**

Fill in the form below to  
place your order.

## Don't bother looking for us on the shelves anymore – it could be a complete waste of time!

No really, you'd be far better off buying up a whole year's worth  
of the only definitive guide to your favourite machine in advance!  
That way you'd be sure of getting your copy each and every  
month right on time with no more scrambling for that last  
remaining copy. Amiga Shopper sells like there's no tomorrow so  
make sure you get your future issues today!

### You get all these benefits when you subscribe . . .

- you save time
- you save trouble
- you guarantee your copy
- you get it delivered

**. . . all for just £14.95!**

FILL IN THE COUPON BELOW OR CALL OUR CREDIT CARD  
HOTLINE ON 0458 74011

**YES, YES, YES!** I WISH TO BECOME A SUBSCRIBER TO AMIGA SHOPPER. PLEASE SEND ME THE NEXT 12 ISSUES.

Please tick the subscription you require

☐ UK **£14.95**

☐ EUROPE **£33.25** (Airmail)

☐ REST OF WORLD **£50.30** (Airmail)

☐ WORLD **£27.60** (Surface)

Please also send me *please tick*

**BACK ISSUES £1.50 EACH**

☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 7 ☐ 8

☐ SAMPLER DISKS 1, 2 and 3 **£1.75**

☐ BINDER **£4.95**

Total amount payable £

NAME  TELEPHONE NO

ADDRESS

POST CODE

METHOD OF PAYMENT *please tick*

☐ VISA/ACCESS

CARD NO

EXPIRY DATE

☐ CHEQUE

Please make cheques payable to FUTURE PUBLISHING LTD and send together with this card in an envelope to the  
following address: AMIGA SHOPPER, FREEPOST, SOMERTON TA11 7BR

# PUBLIC DOMAIN

## GREAT SOFTWARE CAN BE YOURS FOR AS LITTLE AS ABSOLUTELY, TOTALLY, DEFINITELY, VERY NEARLY, JUST ABOUT NOTHING AT ALL. (HONEST)

Owning an Amiga, PC or ST now means that you can buy incredibly good software for little more than the price of a disk. Stunning games, utilities, demos, graphics, word processors and more can be found in Public Domain and Shareware software.

But (but! but!!) where do you start? How do you know which programs are completely wonderful and which ones should be re-formatted instantly? What's the best way to buy PD? And – whilst we're at it – just what is PD anyway?

For these answers and a whole trainload more, you'll need a new magazine called **PUBLIC DOMAIN**.

- Every issue is **packed full of reviews** – PD programs assessed for the Amiga, PC and ST.
- Every review is **screenshotted**. You can see exactly what you'll be buying.
- Every copy of PUBLIC DOMAIN comes with an incredible **program-packed coverdisk** – so you can get the very best PD for your machine every time.
- Every page is in **glorious full colour**, so you'll have unrivalled quality.
- And every edition of PUBLIC DOMAIN is backed by the full editorial resources of **Europe's most successful** news-stand computer magazine company, Future Publishing.

**PUBLIC DOMAIN** comes to you from the company that brings you Amiga Format, PC Format, ST Format, Amiga Shopper, PC Plus, Commodore Format, Your Sinclair, New Computer Express, Amiga Power, Amstrad Action, PC Answers, Sega Power and 8000 Plus; markets leaders all.

**Issue 4 on sale 6th February at £2.95**

**IF YOU WANT OUR ADVICE, BUY IT!**

**INCREDIBLE PACKAGE!  
MASSIVE MAGAZINE  
PLUS PROGRAM  
PACKED COVER DISK!**



# **IF AMIGA SHOPPER HAD ITS OWN SHOW, YOU WOULD EXPECT:**

- a) The latest software**
- b) The fastest hardware**
- c) The best advice**
- and....**



**It'll save you serious money**

# BARG

**WEMBLEY  
MAY 15 - 17**

**YES! I would like to order advance tickets to the Amiga Shopper Show and save £1.05 a ticket.**

Please send me \_\_\_\_ adult tickets at the special discount price of £4.95.

Please send me \_\_\_\_ under-14 tickets at the special discount price of £2.95.

Total payment: £.....

Method of payment (please circle) Access/Mastercard Visa Cheque PO

Credit card number \_\_\_\_\_ Expiry date \_\_\_\_\_

Please make all cheques payable to AMIGA SHOPPER SHOW

Name .....

Address.....

.....  
.....

Postcode .....AS03

**Send this form to:**

**AMIGA SHOPPER SHOW**

**PO Box 2**

**Ellesmere Port**

**South Wirral**

**L65 3EA**

**This offer closes on May 8th.**

# AINS!

Amiga Shopper is dedicated to helping you make the most out of your Amiga. So when we organise a show we do it with you in mind. And we have organised a good one this year, at Wembley from May 15-17. You'll find help and advice from the Amiga Shopper team. You'll find the latest hardware and software on display. And of course you'll find bargains galore. We think the Amiga Shopper Show will be your kind of event. Book your tickets now and save money into the bargain.

**Ring our ticket credit  
card hotline on:  
051-356 5085**



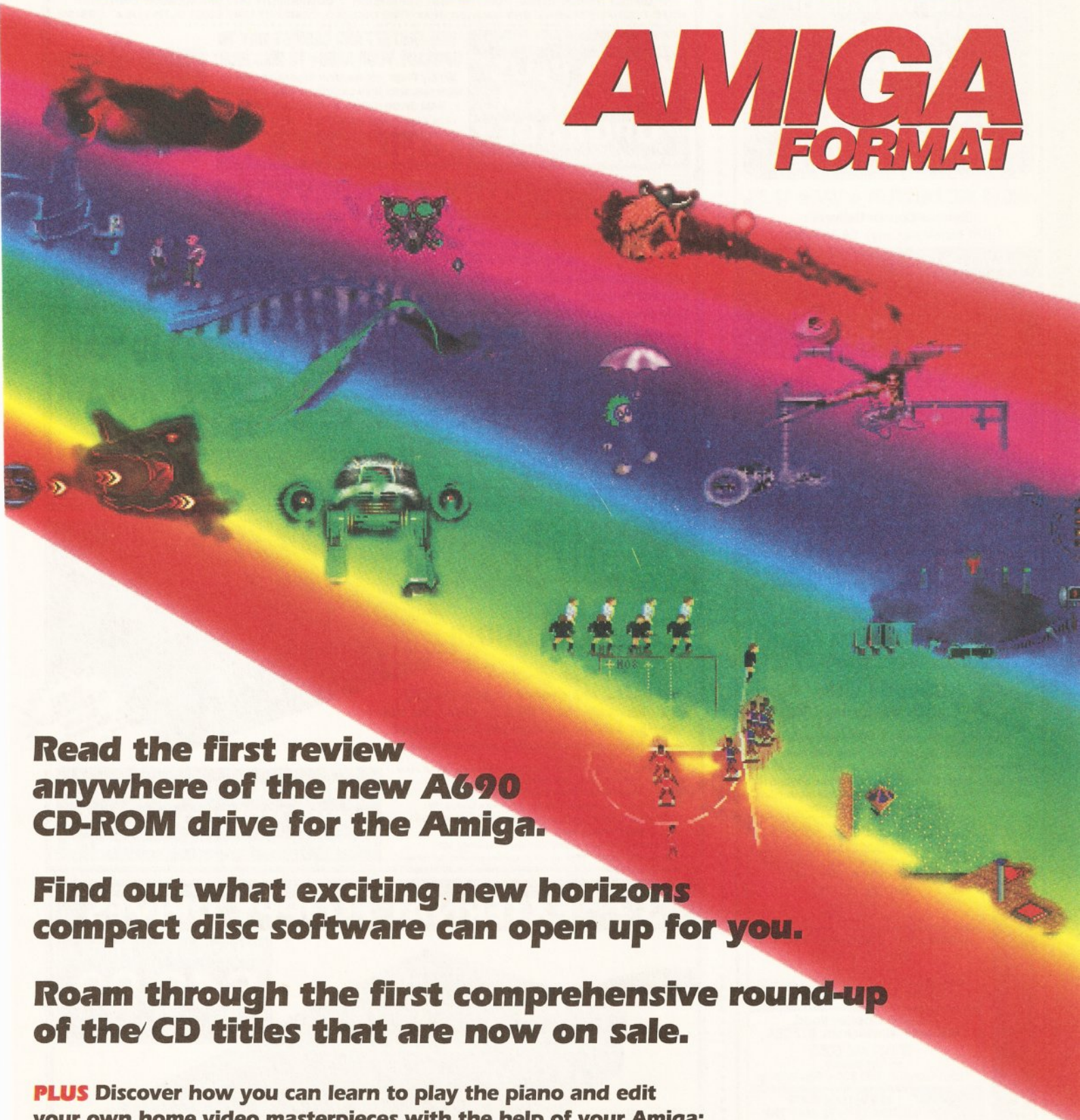
WEMBLEY EXHIBITION CENTRE • MAY 15 - 17

**It'll save you serious money**



# CD is here!

## AMIGA FORMAT



**Read the first review  
anywhere of the new A690  
CD-ROM drive for the Amiga.**

**Find out what exciting new horizons  
compact disc software can open up for you.**

**Roam through the first comprehensive round-up  
of the CD titles that are now on sale.**

**PLUS** Discover how you can learn to play the piano and edit  
your own home video masterpieces with the help of your Amiga:  
and get in on the action as our new series for newcomers  
to the Amiga makes its first tentative steps into using Workbench and the CLI.

**AMIGA FORMAT ISSUE 32 MARCH 1992  
ON SALE THURSDAY FEBRUARY 13TH £2.95**







# You Too Can Program

If you're a beginner, a brief glance through the programming-orientated articles in *Amiga Shopper* has probably convinced you that you don't understand it and that it's all beyond you. Pieces of code look terribly obscure, there's all sorts of talk of libraries and Intuition messages and so on, and there's a horrendous amount of jargon.

In fact, programming is easy. As with anything else, it's a matter of learning the basic principles first. Once you have these under your belt, you'll find yourself solving more and more complex problems without a qualm. Before long it will be time to delve into the innards of the Amiga and start doing some really fancy things.

One of the great benefits of learning to program is in gaining a better understanding of your machine. You'll begin to see how other people's programs work, and to know what is and what isn't possible on the Amiga. You'll feel much more comfortable using the machine, and more confident to boot. And, apart from anything else, programming itself is great fun.

This series aims to take you, gently, from programming ignorance to a state of competence. Most of the principles I'll be discussing are general; they're applicable to any language. It will be necessary from time to time to use short code examples, and in these cases the language used will be Basic.

## GOING WITH THE FLOW

A program is written as a series of statements, representing an instruction to the computer, and usually each is on a separate line. In the normal course of events, each statement is executed in turn, from the start of the program at the top to the end at the bottom.

The order in which the statements are executed is called the 'program flow'. It is possible for the programmer to control the program flow with the aid of 'flow control' language statements – this being one of the main reasons why computer programs can be so powerful. We'll look at flow control in depth later.

## Learn to program like a true professional with a little help from Cliff Ramshaw...

### EASIER THAN SPEAKING

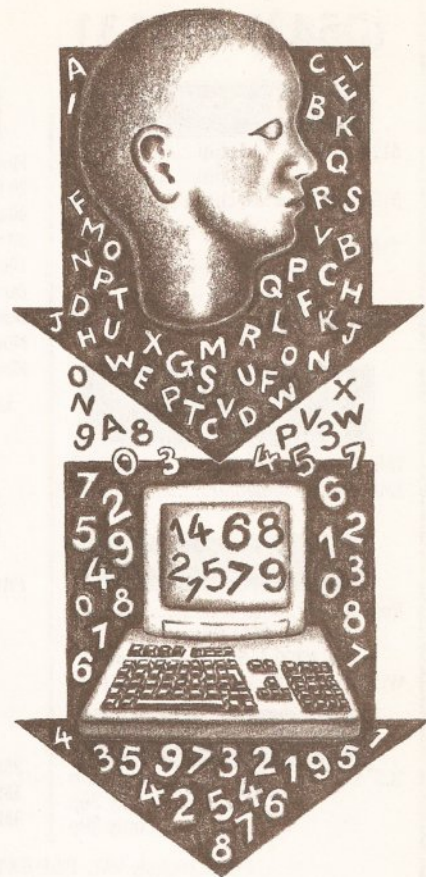
Program languages are very different from spoken languages such as English. For one thing, they have a very much smaller vocabulary. Also while English is rich with potential for metaphor and ambiguity, every statement in a program language must have a single, precise meaning.

Much more care must be taken over the ordering and structuring of statements in a programming language. In English you could say 'I took the dog out' or 'the dog was

taken out by me' and be understood in both cases. With a programming language, there is usually only one correct way to express something. Also, characters such as punctuation marks are of crucial importance and can only be used in specific places for specific reasons.

### OUT WITH THE NEW

Well, it's about time to start writing a program. Most of



## JARGON BUSTING • JARGON BUSTING

**Address** – A number used to refer to a specific memory location.

**Assignment** – The process of storing a value in a variable.

**Basic** – Beginners' All-purpose Symbolic Instruction Code, a language designed to teach people to program.

**Binary** – A number system consisting of two digits (zero and one), useful because it corresponds directly to the electronic signals used in logic circuits.

**Bug** – A mistake in a program.

**Compiler** – A program which translates other programs wholesale into the machine code that the central processor understands.

**CPU** – Central Processing Unit, the controlling part of the computer which follows all program instructions.

**Data** – A computing term for information.

**Denary** – The number system used by everyone, which makes use of ten digits.

**Hexadecimal** – A number system used by programmers which makes use of sixteen rather than ten digits.

**Input** – A Basic statement which takes a value typed by the user and puts it in a variable.

**Interpreter** – A program which translates other programs line by line into the machine code that the central processor understands.

**Machine code** – The set of instructions understood and followed by the central processor.

**Memory** – Set of chips used to store information and programs.

**Operating system** – A program which runs continuously and provides 'house-keeping' facilities.

**Program counter** – A register inside the central processor which keeps track of where the next instruction to be executed is held in memory.

**Print** – A Basic statement which puts information on the screen.

**Register** – A memory location within the central processor.

**Variable** – A number of memory locations grouped together and referred to by name, of use for storing information.

what we discuss here will be very simple Basic, so whether you are using AMOS, Amiga Basic, GFA or whatever, you should have no problems in getting anything to work.

One of the first requirements of your program will be to produce some output on the screen – that way you can easily tell whether or not it has worked properly. The classic beginners' programming example is the 'hello world' program, which, as you might expect if you've been following carefully, prints the words 'hello world' to the screen. The Basic program to do this is shown below:

```
Print "hello world"
```

It's pretty short, isn't it? The 'Print' statement is a Basic instruction to send output to the screen. It can send lots of different kinds of things, but by enclosing the following text in quotes we've instructed it to print the text exactly as it is written.

A program to print two lines of text looks like this:

```
Print "hello world"
Print "how are you doing?"
```

At the end of a 'Print' statement, the Amiga assumes that output has finished on that particular line, and that any further output should be sent to the next line. You can override this assumption by ending the 'Print' statement with a semi-colon:

continued on page 135



# THE OPERATING SYSTEM

If, you may be wondering, the computer is constantly executing instructions, why does it appear to be doing nothing when it's first switched on? In fact a number of programs are running which collectively make up the operating system.

The operating system performs the house-keeping operations of the computer – it makes the disk drives work, reads the keyboard and mouse, displays text on the screen and so forth. It is especially complicated on the Amiga because of the computer's multi-tasking ability. More than one program can be running at once because of a timing trick carried out by the operating system. It devotes a little bit of the processor's time to each programming running in turn – these slices of time are so small that, to the user, everything seems to be running at once, including the operating system which itself is another program running along with the others.

It's not really necessary to know too much about the operating system when writing simple programs, but it's as well to bear in mind that any program you write and subsequently run will not be the only one running on the machine.

continued from page 133

```
Print "hello world";
Print " how are you doing?"
```

will print everything on one line. Notice that there is a space between the quotes in the second line and the first word. As with everything else between the quotes, it will be printed on the screen – it is used so that the two words 'world' and 'how' are not printed as running into each other.

## INCONSTANT SYMBOLISM

Obviously computer programs do more than just print text on the screen. Anything more, though, requires an understanding of variables.

Variables are imaginary containers which hold things: their contents may vary throughout the life of a program, hence their name. Every variable in a program has a name, beginning with an alphabetic character and followed by a number of alphanumeric characters.

A variable is referred to by its name, but what is actually meant by such a reference within a program is the variable's contents, or 'value' – the variable name is merely a symbol used to get at the value.

It's a good idea to choose sensible names for the variables in your programs. Then, when you look back at them, you will have some idea of what sort of values each variable is supposed to be holding.

Note that although just about any word can be used to name a variable, there are some restrictions. For one thing, there is the restriction of length: in Amiga Basic no variable name can be more than 40 characters long. More importantly, a variable cannot have the same name as one of the language instructions. Imagine the confusion if there were a variable in

your program called 'print'.

A variable is a more sophisticated form of memory location. Rather than just being able to store numbers between 0 and 255, variables can store all manner of things.

Exactly what use a variable is to be put to within a program must be stated before it is used. If no other declaration is made, then the variable is usually assumed to be a number variable. Some Basics, such as AMOS, assume that the kind of number is an integer (a whole, non fractional one), and others, such as Amiga Basic, that it is one with a decimal point in it. A distinction is made because although integer numbers don't have the precision of decimal ones (called 'floating point' numbers) they are quicker to use and take up less of the computers

memory, and often the precision of a decimal number is unnecessary.

Variables can have values assigned to them. In Basic, this is done in the following manner:

```
MONEY=100
```

The variable is called 'MONEY', and it has been given a value of 100. This value may subsequently be used in a calculation:

```
pay=MONEY-20
```

which would assign a value of 80 (100-20) to a variable called 'pay'.

As well as subtraction, it's possible to perform addition, multiplication, division, and a number of scientific operations such as sine and cosine. Several mathematical

continued on page 139

# How Computers Work

## A Rough Guide

A rough understanding of how computers work is of real help to the programmer.

At the heart of the computer is a silicon chip called the central processing unit. This is the bit which controls everything else – it is the computer's 'brains' (don't make the mistake, though, of believing that computers can think – they can't). The other main element of the computer is memory where, naturally enough, information is stored. There's a load of support circuitry to aid the passing of information between the processor and the memory (as well as support circuitry to control the screen, keyboard and so forth), but these are the two essentials as far as the programmer is concerned.

Memory is a collection of individual cells, or locations. In the case of a 0.5Mb Amiga, there are over 500,000 of these locations. Each one can store a number between 0 and 255. How this information is used is entirely dependant on the processor. You may have been under the impression that computers can store more than simple numbers. In fact, it's a question of how these numbers are interpreted by the processor. Two numbers can be joined together to form a larger number; a number can be used to represent a letter (so forming part of a word); a number can represent an area of colour on the screen; or it may be used to represent part of a sound sample.

As well as having a number as its contents, each memory location is also referred to by a number. This is known as the location's address. For example, the first location in memory has an address of 0, but it could be holding any number between 0 and 255, say 36. If the processor wanted to know what was in the first location it would ask the support circuitry to provide the contents of address 0 and be presented with the result 36. This would be stored in the processor's own memory, which is much smaller (64 locations in total) compared to the external memory but can be used much more quickly since it is physically part of the same chip. The processor might then add 71 to the number in its own memory, giving the result 107. This result could then be

stored back in location 0.

All operations are performed in this way – the processor's internal memory is used rather like a scrap of paper for performing quick calculations, but all of the important results are stored in the main memory. The processor follows one instruction after another, doing exactly as the instruction tells it. But, you might ask, where do the instructions come from?

As with everything else, the instructions are numbers – with each number representing a specific action that the processor must perform – and as such they must be stored in memory. A collection of these instructions forms a program. Things are beginning to get a little confusing – not only are the results of the computer's calculations held in memory, but so are the instructions which actually describe the calculations. What's going on? So far as memory is concerned, one number is the same as another – it's the processor that makes the distinctions.

One of the processor's internal memory locations (actually, several joined together so that large numbers can be stored) is called the Program Counter. This holds the number of an address in memory. The processor looks at the contents of the Program Counter, and then fetches the information stored in the location whose address is stored in the Program Counter. This fetched information is the next instruction. The processor does what the instruction tells it to, and then adds one to the contents of the Program Counter. It then goes through the whole rigmarole again, getting its next instruction from memory and dealing with it.

These instructions, known collectively as machine code, are very simple. They tell the processor to store things in memory, to retrieve things from memory, to add or subtract things, and other very basic tasks – it takes a lot of these instructions to do anything useful. Anything that the instructions tell the processor to store or retrieve is known as data, which is just another word for information. This can be put anywhere in memory so long as it is safely away from the area of memory where the instructions themselves are stored.

And that is basically how a computer works. The processor just goes on getting its next instruction from memory and doing as it's told until the power is switched off.

# MORE MEMORY POWER!

**NOW YOU CAN ADD UP TO 1.5 MEGS TO YOUR AMIGA AT THE LOWEST PRICES EVER!!**

**UNIQUE FASTMEM/  
CHIPMEM OPTION\***

## PRO-RAM PLUS™

● THE FIRST RAM UPGRADE CARD TO OFFER THE UNIQUE FASTMEM OR CHIPMEM OPTION AT THE FLICK OF A SWITCH\*. NOW YOU CAN HAVE THE EXTRA SCREEN HANDLING ABILITY OF THE A2000 - PLUS LOTS MORE!!

- Simply plug into your Amiga A500 and upgrade to a full 1 Meg in minutes!!
- Features the latest 1 Meg DRams. Low chip count means very low power consumption. No bigger than a Credit Card!!
- Top quality high grade PCB and connector to give unmatched reliability - 12 months guarantee.
- Switch to enable/disable plus "ChipMem" option\* (02 Agnus).
- Now you can run ALL Amiga 1Meg programs.
- Also available with clock/calendar option.
- Easily fitted into the slot under any A500 - does not affect warranty.

**NOW ONLY £24.99** STANDARD VERSION  
**£29.99** FOR CLOCK VERSION

## RAM-MASTER II™

- IF YOU NEED EVEN MORE POWER THEN THE RAM-MASTER II™ IS FOR YOU!!
- Adds an extra 1.5 Megs to your A500 (giving a total free memory of 2 Megs).
- Uses the latest 1 Meg DRams resulting in extra low power consumption.
- Ram-Master II needs Kickstart 1.3 onwards.
- Complete with onboard clock/calendar.
- Top quality connections used throughout.
- Comes with 12 months guarantee.
- Easily fitted into the slot under your A500 - requires no soldering (the cover of your A500 must be removed for fitting "plug in" connector #)

**NOW ONLY £79.99** WITH 1.5 MEGS OF RAM  
**ONLY £39.99** FOR CARD WITHOUT RAM



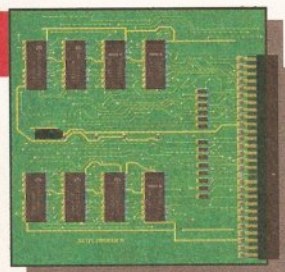
**UPGRADE YOUR  
A500 TO 2 MEGS!**

## ATTENTION ALL A500 PLUS OWNERS

**IF YOU HAVE AN AMIGA A500 PLUS THEN WE CAN OFFER  
A PLUG IN RAM UPGRADE TO BRING YOUR MACHINE UP TO  
A FULL 2 MEGS OF CHIP RAM**

- Simply plug in - no modifications.
- Uses low power surface mount chips.
- Does not affect warranty.

**1 MEG RAM BOARD  
ONLY £79.99**



### HOW TO GET YOUR ORDER FAST!

TELEPHONE [24Hrs] **0782 744707** CREDIT CARD ORDERS

ORDERS NORMALLY DISPATCHED WITHIN 48 Hrs. ALL CHEQUES/POSTAL ORDERS MADE PAYABLE TO...



**DATEL ELECTRONICS**



GOVAN ROAD, FENTON INDUSTRIAL ESTATE, FENTON, STOKE-ON-TRENT,  
ST4 2RS, ENGLAND. FAX 0782 744292 TECHNICAL/CUSTOMER SERVICE 0782 744324

### DATEL LONDON SHOP



222, TOTTENHAM COURT RD,  
LONDON, W1. Tel 071 580 6460

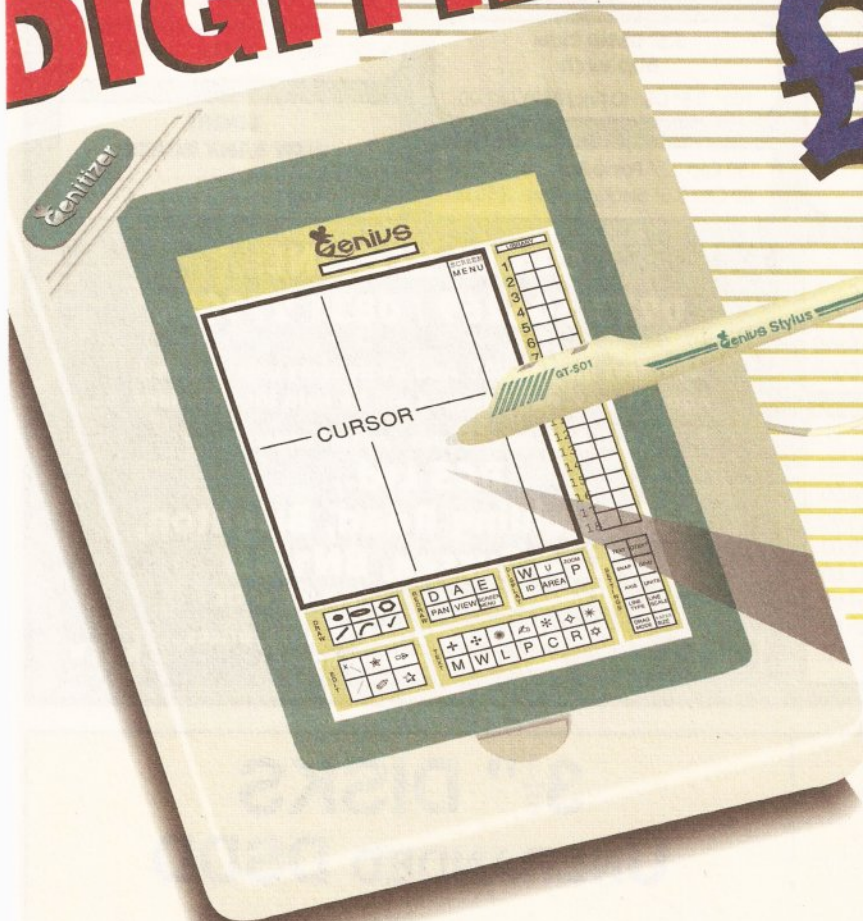
# GENIUS DIGITIZING TABLET

## £129<sup>99</sup>

COMPLETE FOR THE

# AMIGA

ADD A  
PROFESSIONAL  
TOUCH TO YOUR  
DRAWING / CAD  
WORK



Full easy to follow instructions.

This is the input method used on professional systems - now you can add a new dimension to graphics/CAD.

Fast input of drawing by "tracing" is made easy - plus "absolute reference" means you can move around the screen many times faster than by a mouse.

The Genitizer fits in the serial port of your Amiga

500/1000/2000 and "co-exists" with mouse.

Unlike a mouse, the Tablet gives absolute co-ordinates so that tracking and menu selections are possible from the tablet easily.

A pressure sensitive switch built into the stylus tip activates the Tablet overriding the normal mouse input. When you are using the Tablet, you have normal mouse control.

Complete system - Graphics Digitizer Tablet, Stylus, Deluxe Paint Template, Power Adaptor, Test Software, Interface Unit, and Driver Program - no more to buy!

- With the Amiga Genitizer Graphics Tablet you can streamline the operation of most graphic or CAD programs.
- The Genitizer Graphics Tablet utilises latest technology to offer up to 100 dpi resolution at the tip of a stylus.
- Complete 9" x 6" digitizing area plus super accurate stylus combine to enable fast, accurate and easy control.
- Works by "mouse emulation" so the Genitizer will work with most packages where mouse input is the usual method - Deluxe Paint, Photon Paint, CAD Packages, etc. etc.
- Supplied with a template for Deluxe Paint.



**OPTIONAL PUCK**  
As an alternative to the stylus input the Genius Tablet also has an optional four button puck. Ideal for tracing etc.  
**ONLY £29.99**

**DATEL**  
Electronics

**HOW TO GET YOUR ORDER FAST!**

TELEPHONE [24Hrs] **0782 744707** CREDIT CARD ORDERS

ORDERS NORMALLY DISPATCHED WITHIN 48 Hrs. ALL CHEQUES/POSTAL ORDERS MADE PAYABLE TO....



**DATEL ELECTRONICS**



GOVAN ROAD, FENTON INDUSTRIAL ESTATE, FENTON, STOKE-ON-TRENT,  
ST4 2RS, ENGLAND. FAX 0782 744292 TECHNICAL/CUSTOMER SERVICE 0782 744324

**DATEL LONDON SHOP**  
222, TOTTENHAM COURT RD,  
LONDON, W1P 9AF.





continued from page 135

operations may appear on the same line, and in this case it's necessary to have some rules about which operations are performed first, known as the order of precedence. In most languages, including Basic, multiplication and division have a higher precedence than addition and subtraction. Eg, the following assignment to the variable MONEY:

```
MONEY=100+20*10
```

sets it to a value of 300. Because multiplication has a higher precedence than addition, the two numbers to be multiplied are done so before the addition is performed.

The order of precedence can be artificially changed by the programmer with the use of brackets. If the above example was re-written as:

```
MONEY=(100+20)*10
```

then MONEY would end up with a value of 1200. Brackets have the highest precedence of all, and in this case they have forced the computer to evaluate the addition before going on to perform the multiplication.

It's all very well being able to give variables values and to do arithmetic with them, but really we'd

like to be able to see what values they hold as well. Needless to say, this is possible with Basic, and it's carried out with the Print statement, which was introduced earlier in this article.

Before, Print was used with quotes, which meant that everything within the quotes was to be printed literally.

If Print is used without the quotes, but with a variable name,

## Interpreters and Compilers

Programs are written with the aid of an editor – a kind of word processor – and are stored as text files. Each character of a program is represented by a number between 0 and 255. All of the characters, reading from the top left to the bottom right of the program, are stored consecutively. Once in memory, each of its characters is stored in a single location.

In a language such as Basic, each instruction is made up of one or more words, which in turn are composed of several characters. These are not the same as the machine code, mentioned earlier, which the Amiga's processor understands. Before the Basic instructions can be processed they must be translated into machine code by another program.

There are two different approaches to the problem of translation. The first is that employed by an interpreter program such as AMOS or Amiga Basic. An interpreter is a machine code program. It looks at each line in turn of a user's program, works out which instruction the line represents, and executes a set of machine language instructions to perform the desired operation. Once it has done this, it then goes on to the next line in the program and translates it in the same way. If the program is run again, then each of its lines must once again be translated – the interpreter's translation is not remembered, it is only used on the fly.

A compiler translates a user's program wholesale, producing a program which is composed entirely of machine code. This program can then be run independently of any further translation.

then the value held by that variable is output:

```
Print MONEY
```

would output whatever number was currently stored in the variable MONEY. Both variables and text can be mixed, as in the following:

```
Print "The amount paid ↓  
is";MONEY
```

Notice the use of the semi-colon. As before, it instructs the computer to print the next item immediately after the text in quotes, but it also serves another purpose: it acts as a separator between the two things to be printed; its use is necessary if the two items are to be printed with a single Print statement. Any number of items can be included in a single Print statement, up to the line length limit of the particular Basic you are using, but each of them must be separated from the rest.

As well as the semi-colon, it is possible to use a comma as a separator. This has a slightly different effect. Whereas a semi-colon meant 'print the following on the screen immediately after the preceding output', a comma means 'move to the next tab position across the screen, and print the following'.

Tab positions are ways of organising the screen into vertical columns. In Amiga Basic there are six, with the first starting at the left-hand column of the screen and the second being fifteen spaces along to the right. Commas are useful as separators when you are printing out lists of numbers, since they allow information to be tabulated in a neat way onscreen.

### GETTING IT IN

Now that you know how to mess around with information inside the computer and how to print it out, you might be interested to know about getting it into the computer in the first place. This is usually handled in Basic with the 'Input' statement, followed by the name of a variable:

```
Input MONEY  
Print "The amount you ↓  
entered is";MONEY
```

When a program is running and the computer comes across an Input statement, it stops what it is doing and waits for the user to type a response. In AMOS and Amiga Basic, it lets the user know that it is waiting by printing a question mark. The user then types in the required input, presses [Return], and the program carries on, with the value the user typed being stored in the variable following the 'Input' statement.

continued on page 140

## Input and output

Just about every program requires data to run properly. Programs can be viewed as machines which take a lump of data and convert it into another, different lump of data. An adding program is a good example: it takes two numbers as data and produces another number. The first two numbers are known as the program's 'input'; the last is the program's 'output'.

For any important application, the data is actually more important than the program. People want programs to do particular jobs – they are not interested in a program per se, only in what it will do to their data. Obviously then, when writing a program it's important to consider what kind of data it's going to need, and in what form, and what form its output will take.

One thing to realise (although you probably already have, if you've used an Amiga for any length of time) is that programs are not infallible. A program can have a mistake (or 'bug') in it, causing it to behave erratically and perhaps give false results; or it may simply be given incorrect input data, in which case its output will assuredly be wrong also. This latter case is summed up by a famous old computer professional saying, "garbage in: garbage out."

We'll discuss different forms of data in more detail once a few of the basics of programming have been dealt with.

continued from page 139

Giving the user nothing more than a question mark when your program expects input can be confusing, so it's a good idea beforehand to print a little message telling them what your program expects:

```
Print "Please enter the ↵
amount of cash"
Input MONEY
Print "The amount you ↵
entered is";MONEY
```

That's one way of doing it, but there is another, slightly shorter way. It is possible with Basic to use the 'Input' statement like a 'Print' statement, and actually print text to the screen as well as getting the value for a variable from the user:

```
Input "Please enter the ↵
amount of cash? ";MONEY
Print "The amount you ↵
entered is";MONEY
```

In this case, the message to be printed must be in quotes – it cannot be a variable – and the variable to which the value is to be input must follow the quoted message and a separator.

In AMOS, the separator must be a semi-colon, and no question mark is printed as with an ordinary 'Input'.

In Amiga Basic, either a semi-colon or a comma is used: in the case of a semi-colon, a question mark is still printed (so two would be printed in our example above); in the case of a comma the behaviour is like AMOS' 'Input', with no question mark being printed. As you can see, the differences between the various versions of Basic are many and subtle.

### VAT AND ALL THAT

There's just space this month to introduce a very simple program. The

following program will ask the user for the price of an item, and then print this out along with the price including VAT:

```
Input "Enter cost price ↵
of item ";COSTPRICE#
SELLINGPRICE#=COSTPRICE#
#+COSTPRICE#*17.5/100
Print "Cost price"," ↵
Selling price"
Print COSTPRICE#, ↵
SELLINGPRICE#
```

The first line prints the message in quotes to the screen and then gets a value from the user which is stored in the variable COSTPRICE.

Note the hash signs after the variable names. This is for AMOS, and it tells the computer that the variables are to hold numbers with decimal points. The hash is not needed for Amiga Basic.

A value for SELLINGPRICE is then calculated by adding 17.5% (the asterisk means 'multiply' and the slash means 'divide', so multiplying by 17.5 and dividing by 100 calculates 17.5%) of COSTPRICE on to the unmodified value of COSTPRICE.

Next, two table headings are printed out, with a comma in between ensuring that the second heading starts at the next tab position onscreen. Finally, the values of the two variables are printed out, again separated by a comma to tab them across the screen beneath their respective headings.

Obviously the program is a little simplistic, but it will do for a start.

### COMING NEXT MONTH

Next month we'll get stuck into loops and your programs will start to look much more sophisticated. Until then, experiment with what you've learnt so far – you never know what you might discover. **AS**

## Binary and its chums

Sooner or later in your programming endeavours you're going to come across binary arithmetic. An understanding of what it is will be an asset in such a situation.

Binary is a way of representing numbers. Normally, we represent numbers in what is termed 'denary' form – everything is based around the number ten, presumably because we have ten fingers and counting evolved in this way. We have digits from 0 to 9; if we want to represent numbers higher than 9, then we put a 1 in the 'tens' column and a zero in the 'units' column, giving us the number 10. Higher numbers are achieved by increasing the units, until 9 is exceeded once more, at which point the 'tens' column is increased and the 'units' column goes back to zero – the result is the number 20. In a similar way, larger numbers have a 'hundreds' (ten tens) column, a 'thousands' (ten times ten times ten) column, and so on. We learnt this so long ago at school that it's now second nature – we don't normally have to think about it at all.

Binary is based around two digits instead of ten. The reason this is useful is that a computer works by a series of electronic switches, each of which can be either on or off. It's a simple matter to interpret 'on' and 'off' as two digits. In binary, the two digits are '0' and '1'. For higher numbers, a column to the left must be added, just as we add a 'tens' column for denary numbers above 9. In the case of binary, though, the column is a 'twos' column. With a two digit binary number, the highest value that can be represented is 3, and it is written as 11. This means that there is 1 '2' and 1 '1' – a total of three. Another column can be added to the left to give 'fours' (two twos), and a further one to give 'eights' (two times two times two).

As you can see, it takes a lot of binary digits (known as 'bits') to represent even relatively small denary numbers. In the Amiga's memory, eight binary digits are grouped together to hold the contents of each location. Consequently, numbers between 0 and 255 can be stored there, since 255 is the largest value that can be represented with eight bits.

Another number system often used in programming is hexadecimal. This is based around 16 digits – 0 to 15. Anything written in the column to the left represents the number of 'sixteens' in the number. Although the 'units' column of a hexadecimal number may hold a value higher than 9, it would be impractical to write this as an ordinary number with two digits – it would be impossible to tell whether the left-hand digit referred to 'sixteens' or whether it was part of the units. Instead, units between 10 and 15 are written as the letters A to F.

Programmers often use hexadecimal because it is a much more convenient and easier to read notation than binary, yet translation between the two is relatively simple. Here's a table comparing the binary, denary and hexadecimal notations for a set of numbers:

Comparison of denary, binary and hexadecimal numbers

DENARY	BINARY	HEXADECIMAL
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10



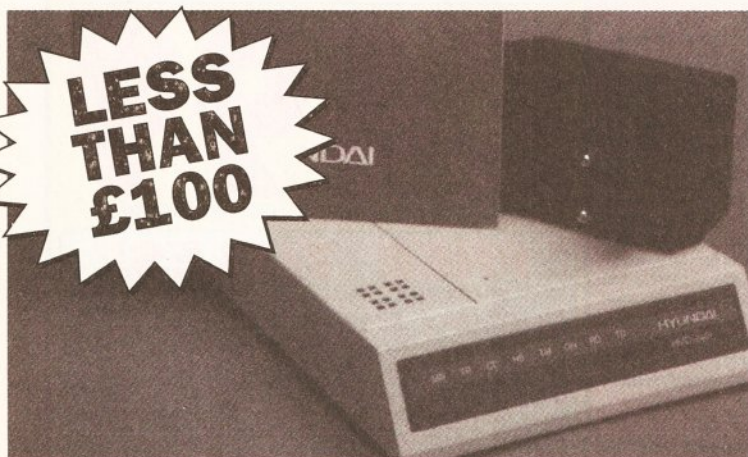
# AMIGA SHOPPER

**SPECIAL OFFER**

## Get on line with our Amiga Shopper modems

### HYUNDAI HMD2401

**LESS  
THAN  
£100**



This excellent budget modem (RRP £129+P&P) is capable of respectable speeds - up to 2,400 baud. It supports the V21, V22 and V22bis protocols. To use it you'll also need a serial cable, and of course some comms software!

**Only £99.95**

### CONNECTIONS M5024/5

**SAVE  
£180!**  
RRP £469+£10P&P



If you want higher speeds, the Connections Interdial M5024/5 includes MNP4 error correction and MNP5 data compression, giving effective speeds of up to 4,800 baud. Again, you'll need comms software and a serial cable.

**Only £299.95**

**I'd like to go on-line.** Please send me a modem at your special discount prices:

☐ Hyundai HMD2401 at £99.95 including VAT and P&P

CODE:AS010

☐ Connections Interdial M5024/5 at £299.95 including VAT and P&P CODE:AS011

Total: £.....

Method of payment (please ring) : Visa Mastercard Cheque PO

Make all cheques payable to **Future Publishing Ltd**

Card No: \_\_\_\_\_ Exp date: \_\_\_\_\_

Name .....

Address .....

.....

.....

Postcode .....

**Send to:** Future Publishing, The Old Barn, Somerton, Somerset  
TA11 7BR

# Pretentions on the PC

**The KCS Power PC board's long awaited colour EGA/VGA upgrade has finally arrived. Mark Smiddy exclusively examines a dream come true**

supplied: one for standard machines, one for machines with extra RAM and another optimised for the 68020/30 CPUs. Auto sensing installation software automatically ensures you boot with the correct one. The PC Preferences (configuration) software can now be launched from Workbench – far more friendly than the original method.

## MORE MEMORY

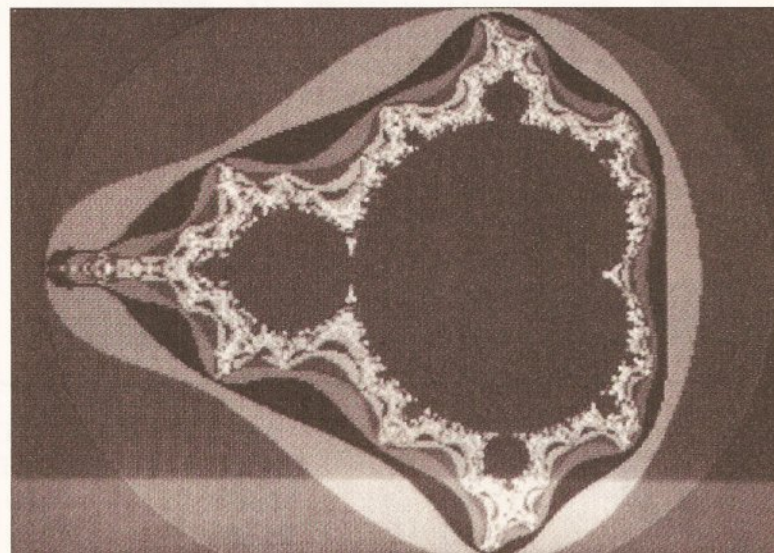
In addition, the Amiga side of things includes another little program which creates an MS-DOS utilities disk. This obviously saves money for everyone when it comes round to sending out the updates, since only one disk has to be duplicated. The software's front end is a little crude but it works, and you only need it once. The disk created contains no less than ten utilities for use with the

PC side of things – including the clock and mouse drivers, plus a 200K PC-reset proof RAM disk.

More notable though are KCSEMS.SYS and KCSXMS.SYS – two drivers for the long awaited Extended (XMS) and Expanded (EMS) memory system. In one fell swoop, these two little programs solve what some have seen as a major flaw in the original software. Practically speaking, these will allow you to access up to 16Mb extra memory in PC mode; allowing massive Lotus 1-2-3 spreadsheets and effective use of Microsoft Windows. KCSXMS.SYS replaces the MS-DOS HIMEM.SYS which only works on 80286 type machines.

## GRAPHICS UNLIMITED

This is the bit we have all been waiting for, the high-resolution colour graphic support in EGA and VGA. The



Another example of the ubiquitous Mandelbrot set, this time produced by the popular PC public domain program *Fractint* in the Power PC Board's 16 colour VGA mode

"If you've just bought an Amiga, then you may ask why you would want to turn it into a PC. I'm explaining just what the KCS Power PC board can offer you."

Mark Smiddy

It's a sad fact that most PC emulations – and you can include machines such as ST and Archimedes – only go as far as emulating IBM's colour graphics adaptor (CGA). There's a good reason for this: the later standards EGA and VGA create a greater demand on the emulation; and therefore the machine performing it.

## POWERBOARD V3

To some extent emulating some modes available on these cards is either impractical or plain impossible because of the physical resolution employed. Vortex boasted a monochrome subset of E/VGA with the last edition of ATOnce; but this proved to be more trouble than it was worth. Almost a year later, KCS has finally released a 16 colour E/VGA solution that promises to be much more compatible. (If you are not yet familiar with the KCS Power PC Board, further details are given in the Beginners Section on page 145).

The most notable change in the new software is that the board is now started from a Workbench icon and not by a direct boot disk – making hard disk installation possible. Floppy disk users have an option to convert to a bootable disk if they wish. Three versions are

## JARGON BUSTING • JARGON BUSTING

**CGA** – Colour Graphics Adaptor. 4 colour low-resolution graphics board.

**EGA** – Enhanced Graphics Adaptor. 16 colour low or medium-resolution graphics board.

**EMS** – Expanded Memory System. Memory located above the normal DOS limit of 640K up to a maximum (usually) of 384K. EMS co-devised by Lotus, Intel and Microsoft (hence LIM) is a paged system, so although you can have very large amounts, only small (384K) chunks are available at any one time. How this affects you depends very much on the software.

**MCGA** – Multi-colour graphics adaptor. 256 colour low-resolution graphics board.

**MDA** – Monochrome Display Adaptor.

**VGA** – Video Graphics Array. 16 or 256 colour, high-resolution graphics board.

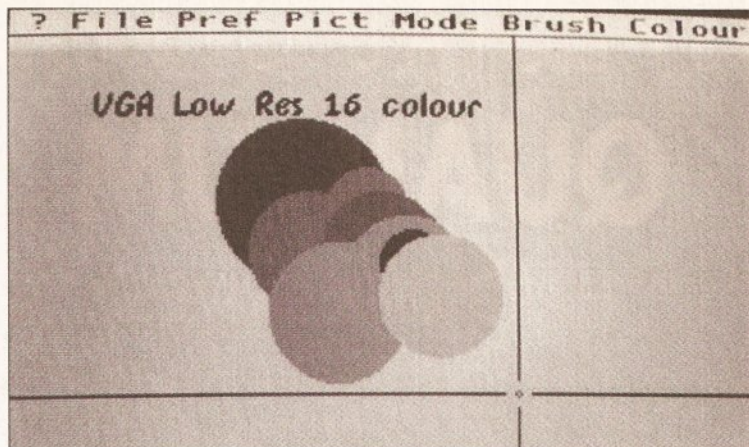
**XMS** – Extended Memory System. Memory located above the 1Mb limit. Used for large RAM disks usually – but only available on 386 processors and above.

task of emulating these modes is by no means as easy as you might think – since PCs have a wide range of graphics modes that must be catered for. ATOnce owners are stuck with mono EGA and VGA which doesn't work very well; so the acid test must be to see how well KCS has addressed the problem. The answer is well, though not perfectly.

In text modes, only the normal 25 line display is supported: there's no support for the 43 line EGA or 50 line VGA modes. Not a lot of programs actually support these, but it would be nice to have the option; although it must be said, Amiga hardware restrictions are partly to blame here. Speedwise, there is little to tell between a real XT (or an AT for that matter) and the emulator. In many tests the video BIOS is actually faster than a real PC in text mode. At this point, it's probably worth mentioning a curious new text mode

continued on page 145





No stranger to Amiga users, this is the PC version of *Deluxe Paint*, which runs at a pinch with the Power PC board emulator

continued from page 143

which allows the screen to be stretched to double its normal width. The screen displays just 40 characters, but the software thinks the screen is still 80 columns wide. A simple keypress scrolls around the display. A nice touch for the visually impaired, I feel.

## FLICKER OFF

Graphics fair slightly less well. As far as I could determine, all the standard 16 colour modes are supported – some using interlace, others not; a full list appears elsewhere. However, although most programs ran perfectly under CGA, a few (notably *As-Easy-As 4*) failed to work properly, if at all, in EGA and VGA. The resultant output ranged from usable (*As-Easy-As*) to totally garbled (*Cadaver*). KCS is aware of the problems and is determined to correct them soon. Based on my previous experience, I believe the company will too. You should also note, due to the type of emulation the aspect ratio of some screen modes gets a little distorted.

Interlaced screens are used whenever the software calls for a display with more than 256 vertical lines – say 640x480 16 colour VGA mode. Interlace, and the flickering screen, is a fact of life but KCS has made things a little easier on the eye with an anti-flicker mode. In colour modes a simple keypress selects an interlaced or low-res display. The full interlaced screen is correctly drawn even when flickering is switched off. The anti-flicker for the monochrome modes is slightly more sophisticated: no less than 17 levels are available at the touch of a button.

## SPEED TESTS

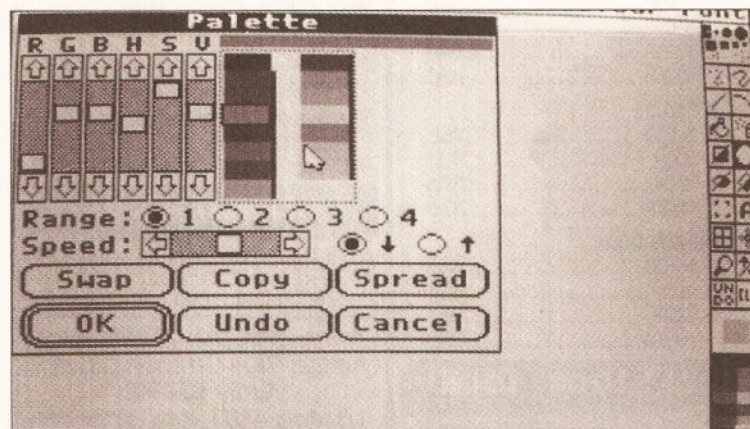
As I have said before, I am not a big fan of benchmark tests – because the only thing you can test with a benchmark is the benchmark test. Even the highly respected Landmark couldn't make its mind up whether the system was running like a 8MHz or 6MHz AT and the Norton index

result came in at 5.1 – faster than the original version 2 software. (Those with a thirst for figures will enjoy the comprehensive list I've compiled here.) My personal preference is to test the system with the software you want to run on it, and then decide.

Text emulation is excellent, and a surprising number of PC programs – like Central Point's *PC Tools* – use nothing more than flashy text displays to achieve the desired

result. These programs run without a glitch – so you can run *DBase IV*, Turbo BASIC, Pascal, C, and so on just as if they were running on a real XT. (The mouse emulation comes in surprisingly handy here; and since

runs – at a pinch, as do *GEM*, *GEOS* and *Windows*. But these systems are not really designed for XTs anyway, so you can't expect the Powerboard to make them do loop-the-loops. Games are graphics and sound



The palette requester for *Deluxe Paint* running in 16 colour VGA mode. Again it's the PC version of the software

the Amiga already has a mouse, there's no need to rush out and buy one for the PC.)

Graphics intensive programs are a whole new ball-game. *Deluxe Paint*

intensive things too; although most will work in icky CGA (very well too, all things considered) EGA and VGA are a dead loss – so you can forget

continued on page 147

## BEGINNERS

### What is the Power PC board?

It's a 11MHz IBM PC-XT clone on a simple plug in card that fits all Amigas. 1Mb of memory is available to the PC side, 512K is added to the Amiga. A special version for the A500 Plus makes a full 1Mb available, taking the machine to 2Mb CHIP RAM; and 1Mb of FAST RAM on the A1500 and above. The BIOS is based on a ROM from Phoenix Software.

Hardware emulation includes: one serial port (COM1 or COM2); one parallel port; a mouse and joystick driver. Screen modes for Hercules monochrome, MDA, plus partial colour EGA and VGA.

### So what's a PC?

A personal computer. When the PC was introduced it created a whole new market. Previously computers were massive affairs housed in purpose-built rooms. Each operator would have their own terminal (keyboard and VDU) but no access to the CPU itself.

The advent of the personal computer revolutionised all that. For the first time ever, computers became desk-sized single user affairs. Since the marketing muscle of IBM came to bear the term, PC has become synonymous with one particular breed of these machines – the IBM PC.

### So what's an IBM PC?

The term PC and IBM PC are very often confused – even by the people who sell them. An Amiga is a PC, as is the Mac – they are all personal computers. Most people regard a PC as being any computer which is based around an Intel 8088 (or compatible) CPU and is capable of running a version of MS-DOS. The range starts with the 4.77MHz 8088-based systems – called as PC-XT clones – rising to the powerful 33Mhz 80486 (aka 486) systems.

Between the two extremes (rising in power) are



## BEGINNERS START HERE

## BEGINNERS

the 8086, 80286, 80386SX, 80386DX and 80486SX – each with a variety of different clock speeds. These days, XT-type systems run at 8 or 10MHz; AT-type systems (based on an 80286 or higher) range from 8MHz to 33MHz and are getting faster all the time. As a benchmark, the 7.14MHz Motorola MC68000 found in most Amigas is roughly equivalent to a 8MHz 80286.

What makes the IBM system clever though, is its modularity. Just about everything you add to the system is plugged into a "slot" in the motherboard. Therefore, you can change just about any aspect of the hardware by exchanging or adding a card.

### I'm happy with my Amiga, what good is a PC?

The PC comes into its own when you look at business applications. It will allow you access to top-selling titles such as *Windows 3*, *Lotus 1-2-3*, *Aldus Pagemaker*, *Microsoft Word*, *DBase IV* and so on. To get the best from any of these you'll need a good graphics adaptor too. CGA is not good enough and EGA is really the minimum; and that's what makes the latest edition of the Power PC board so exciting.

### What are CGA, EGA, VGA and MCGA?

CGA (colour graphics adaptor) was IBM's first attempt at a graphics card – one graphics mode (320x200) is offered in a choice of two sickly palettes. Later came EGA – this improves the situation with three 16 colour modes: 320x200, 640x200 and 640x350 with a 64 colour palette. VGA adds 640x480 to the standard EGA modes and a palette of over 260,000 colours. MCGA is the 320x200, 256 colour mode of VGA. Third party cards offer an even more bewildering variety of modes. However, it's worth mentioning that on a "real" PC these modes all fill the screen; this is not true of the current batch of emulators.



continued from page 145

about *Wing Commander 2*. Games players will enjoy the new features such as pause mode and speed adjustment though.

## SLOT IT IN

Perhaps the Powerboard's biggest shortcoming was its inability to run in Commodore's premier machines, that is, those above the A500. This has recently been addressed in the form of a simple plug-in adaptor board. (A similar remedy to the one offered for ATOnce.) The original board simply slots into the adaptor, which in turn drops into a vacant slot in the host machine; including an A3000. Therefore, if you decide to upgrade to an A2000 machine later on, you can still enjoy the KCS

emulation without having to resort to Commodore's own mediocre offerings or the ATOnce.

Interestingly enough, owners of these machines can enjoy other benefits too. Since the emulation is software based, the added speed and power of, say a 25MHz 68030, speeds up many functions. This is most noticeable in graphics modes, games such as *Flight Simulator 4* and *Lemmings*. Also, Hercules and the high-resolution E/VGA modes are flicker-free.

## CONCLUSIONS

When it comes to PC emulation, the KCS card has always been the one I have unreservedly recommended: in terms of value for money, ease of use and compatibility it is still the best. Version 3 was a long time

### Those speed comparison ratios in full

KCS Power PC board relative to: 4.77Mhz XT 8Mhz AT

#### RAW TEST (A500)

Instruction Mix	2.6	0.7
128K NOP loop	3.1	1.3
Do nothing loop	2.2	0.8
Integer add	2.9	0.6
Integer multiply	4.3	0.5
Sort and move	2.8	0.7
Prime sieve	2.6	0.7
Float mix	3.4	0.7
OVERALL (mean)	3.0	0.8

#### ACCELERATED TEST 68030 25MHz

Instruction Mix	2.5	0.7
128K NOP loop	3.4	1.4
Do nothing loop	2.5	0.9
Integer add	3.3	0.7
Integer multiply	5.0	0.6
Sort and move	2.8	0.7
Prime sieve	2.9	0.8
Float mix	3.4	0.7
OVERALL (mean)	3.2	0.8

#### ACCELERATED TEST 68030 25MHz

DOS disk access (drive C)	5.0	1.9
File access (small records)	2.3	1.3
File access (large records)	4.1	1.7
OVERALL (mean)	3.8	1.6

#### RAW TEST (A500)

Direct screen access	9.7	7.6
Bios teletype no scrolling	4.4	1.9
Bios teletype with scrolling	5.7	2.7
OVERALL (mean)	6.6	4.1

#### ACCELERATED TEST 68030 25MHz

Direct screen access	11.6	9.1
Bios teletype no scrolling	8.6	3.6
Bios teletype with scrolling	23.4	11.2
OVERALL (mean)	14.5	8.0

Conventional memory read	3.2	0.7
Conventional memory write	3.2	0.7

## A Guide to Video Compatibility

Mode	X x Y	Type	Results
0	40 x 25	Text	Pass
1	40 x 25	Text	Pass
2	80 x 25	Text	Pass
3	80 x 25	Text	Pass
4	320 x 200	4 colour CGA	Pass
5	320 x 200	4 colour CGA	Pass
6	640 x 200	2 colour CGA	Pass
7	80 x 25	Monochrome (Hercules)	Pass
13	320 x 200	16 colour EGA	Pass
14	640 x 200	16 colour EGA	Pass
15	640 x 350	Monochrome EGA	Pass
			(Interlace)
16	640 x 350	16 colour EGA	Pass
			(Interlace)
17	640 x 480	2 colour MCGA/VGA	Pass
			(Interlace)
18	640 x 480	16 colour VGA	Pass
			(Interlace)
19	320 x 200	256 colour MCGA/VGA	Fail
P.C.G.			Fail
Split scrn			Fail
Blinking			Fail
Highlight			Pass
Reverse			Pass
Pan test			Pass
Palette			Pass

coming, but it has been worth the wait. The bugs (buggettes really) are being addressed but hardly detract from what must still be the best value-for-money upgrade for the Amiga yet devised. Hard disk owners should still check for compatibility, because although the Powerboard works with most systems it does not support every one - UK suppliers, BDL, will furnish the information at the cost of a phone call.

**"It costs less and does many things better than the real thing."**

Existing users will probably already have their copies by the time they get to read this since KCS has agreed to send free updates to all registered users; unusually generous for a manufacturer - it's a real pity there aren't more like them. Bitcon also must be complimented on their excellent helpline service, one of the best I have come across.

The Powerboard will never replace a dedicated PC, but for occasional use you won't find a better system at the price. What next? I think everyone wants 256 colour MCGA/VGA but that's a bit of a pipe-dream given Amiga hardware. If anyone is going to crack that one, my money's on KCS. **AS**

## SHOPPING LIST

**KCS Power PC Board**  
for board and software only .....£189.95  
inc of MS-DOS & GW Basic.....£219.95  
for A1500/2000/3000 adaptor.£74.95  
by Kolff Computer Supplies BV  
Dordrecht, Holland

UK importer: Bitcon Devices Ltd,  
88 Bewick Road, Gateshead,  
Tyne and Wear NE8 1RS  
☎ 091 490 1919

## CHECKOUT KCS Power PC Board

**Stability** ●●●●●  
Can't find fault with it here.

**Features** ●●●●○  
Even more options than the original.

**Documentation** ●●●●○  
Getting better - new manual expected soon.

**Compatibility** ●●●●○  
EGA/VGA is a little shaky in some modes.

**Speed** ●●●●○  
Excellent, by far the fastest XT I have ever used.

**Value** ●●●●●  
Costs less than the real thing, and does many things better.

**Overall rating** ●●●●○  
The best PC emulator around, and it keeps getting better.

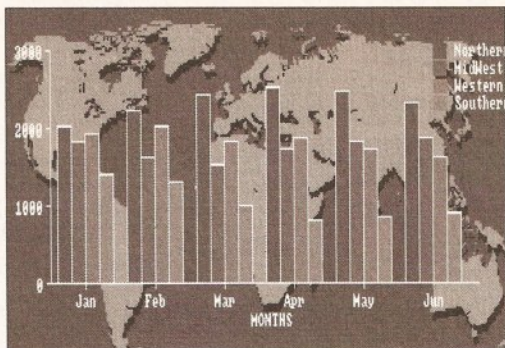












IFF images can be used as a backdrop to any graph and produce interesting results

**S**preadsheets like *Advantage* have offered graphing functions for years, but *Bars&Graphs* is the first dedicated graphing program for the Amiga. Is it any good?

Most of us associate charts and graphs with school mathematics lessons, but they can also be jolly useful things if you use your Amiga for any one of the many forms of computerised presentation. Whether you're a whiz with a genlock, a master of DTP or a high-flyer in business presentation, being able to get complex statistical information across to your given audience is an art form that few have mastered.

Drawing up a decent graph is easy using a program like *DPaint*, but it's hardly the most enjoyable task at the best of times. What's more, if

you're dealing with a complex set of numbers, hand-drawn graphs have a habit of being imprecise. This is where *Charts&Graphs* comes in. It's dedicated to producing attractive charts and graphs in different formats. It does most of the work for you, so put away *DPaint* now!

### TECHIE TWADDLE

I'm one of those people who likes to dive straight in and get started with a program, but soon discovered I'd have to sit down with the 230 page manual to see anything even slightly resembling a graph on my screen. I'm not calling it unfriendly, but it's hardly the easiest program to use.

The *Charts&Graphs* manual has one major problem – it was written by the programmer. On the whole he does a good job of explaining the program's operation, but he manages to fall into the trap that most programmers do – namely, techie twaddle. We all like to know a little about how programs work, but I hardly think that most users want to know the reasoning behind the programmer's decision to use PUBLIC memory instead of CHIP RAM for the requesters!

Thankfully he does include a couple of tutorials which take you through the process of creating charts. It really is worth following these tutorials before you go it alone. By the time you've worked through them, you should have grasped the majority of the features!

### SPREADING IT ABOUT

When *Charts&Graphs* loads, you're presented with a screen with more than a passing resemblance to a spreadsheet. It doesn't have all the high power number-crunching features you'd expect in a spreadsheet, but the spreadsheet way of working has been brought across. It's a sensible move – after all, most computer users should have a basic knowledge of spreadsheets, so it should be fairly simple to use.

Like a real spreadsheet, your graph data is entered into *Charts&Graphs* by typing each set of numbers into their own columns. You're not restricted to a single set

of graph data, so if you're feeling extravagant, you can let your imagination run wild. Bar some pretty basic addition and subtraction facilities, *Charts&Graphs*'s spreadsheet interface is a little lacking in features – it would have been nice to have a few of the more basic spreadsheet functions on tap.

### CRASH AND SPURN

Once your data is entered, you must save your chart data off to disk. *Charts&Graphs* managed to crash on me more than a couple of times, so I wouldn't trust it with anything more than the simplest of graphs – there's nothing more frustrating than having to type in numbers from scratch (it's bad enough the first time!).

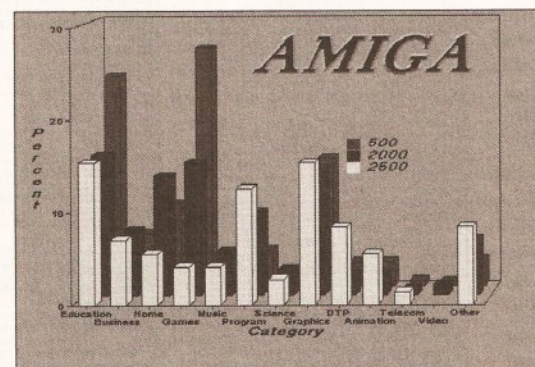
controls range from the tweaking of the colour palette, right up to 3D extrusion and the inclusion of IFF clip art. Indeed, IFF support is one factor that makes the program shine – a backdrop created in a paint package can be draped behind your graph by loading a complete image whilst individual IFF brushes can be loaded and used as the rendering graphic for bar and column charts.

### POSTSCRIPT POWER

*Charts&Graphs* is a pretty well endowed program when it comes to saving and printing graphs. You can save your graph either as a standard IFF image (handy for other presentation programs) or even as a Postscript file for inclusion within

DTP programs like *ProPage*.

Postscript support is extended to printing, allowing you to dump your graph out to anything from a Laser to an Imagesetter with the graph being printed at the highest resolution of your printer. For those of you with slightly more humble printers, *Charts&Graphs* also supports the usual preference printers.



With a single click of the mouse, your graphs can be brought into the third dimension

Selecting 'Graph' from the pulldown menus takes you into the *Charts&Graphs* graphing module – the part of the program which does all the hard work. At the simplest level, there are 19 types of graph to choose from – four line charts, six column charts, five bar charts and four pie charts. Selecting one brings up a requester which asks you which of the columns of data you created earlier is to be used for labelling and which are to be graphed. Once you've told it this, *Charts&Graphs* puts its thinking cap on and after a few seconds your chart is displayed onscreen in all its pixelised glory.

If you're not quite happy with the appearance of your graph, *Charts&Graphs* gives you extensive control over its appearance. These

### CONCLUSION

*Charts&Graphs* is the work of a single programmer and it shows. It can produce impressive results, but lacks the professionalism that should separate it from other graphing programs. The programmer seems to have spent too much time making the requesters look pretty rather than making the program smoother in operation.

It is hindered by its ability to crash when least expected. I thought this might be a problem with my machine, but it continued to crash on two other Amigas – unacceptable when you're paying £50!

*Charts&Graphs* has potential, but this release fails to realise it. You're probably better off paying for something like *Advantage*. **AS**

## SHOPPING LIST

**Charts & Graphs** ..... £49.95  
by TRSL, PO Box 94263,  
Las Vegas,  
Nevada 89193-4263  
USA

## CHECKOUT CHARTS & GRAPHS

### Features

Despite crashes and a slow user interface, a wealth of graphing functions.

### Ease of Use

Getting used to the program takes some time, but perseverance can reap rewards.

### Documentation

Apart from the occasional techie twiddle, the manual is actually very well written.

### Speed

Chugs along at an acceptable pace, but start to throw complex data at it and things can slow down considerably.

### Price Value

Cheap for a commercial product, but still very rough around the edges.

### Overall rating

Locked away by some lousy programming is a gem of a program – it's a shame that the damned thing keeps crashing!





The quality of *TurboPrint* is more than obvious – just compare these two grabs. The top one was produced using the standard Workbench drivers and the bottom one was printed using *TurboPrint*

# Prints Charming

*If the quality of your printouts isn't up to scratch, then Turbo Print Professional could be for you. Jason Holborn wasted lots of paper trying it out*

**T**he Amiga's printer drivers are good as they stand, but I've always suspected that there was still room for improvement. IrseeSoft's *TurboPrint Professional* claims to do the job, but is it any good? Let's find out....

If there's one aspect of the Amiga which causes frustration, it must be getting it to work with printers. Even once you've convinced the Amiga that it ought to talk to your printer, the results are often disappointing, to say the least.

*TurboPrint Professional* may be able to help you put. It is a vastly enhanced version of Irsee's *TurboPrint* utility, launched about 4



drivers for Panasonic, Epson, Seikosha, NEC and Brother printers. The *TurboPrefs* program is split into four subsections.

quality – and what a difference! Having suffered the standard drivers for years, I was surprised by the increase in print quality I obtained.

## HOT STUFF

*TurboPrint* isn't one of those programs that just replaces the systems software and nothing more – it does actually give you a few extra facilities. These are accessed via 'hot keys' (pressing a combination of keys simultaneously) and can be used within just about any program. There are four of these: hardcopy screen, hardcopy area, save screen and save area.

The two hardcopy options enable you to immediately dump just about any screen that was opened under Intuition to your printer. Hardcopy area enables you to specify which part of the screen is to be printed by enclosing it in a rectangle. The two save options do the same job but the results are saved as IFF files.

Finally we have poster mode. This enables you to print any screen across several sheets of paper which can be joined together afterwards to form a massive poster with a maximum size of 8 by 7 sheets of A4. I couldn't quite face waiting for 56 sheets to churn out of my printer, but I did try it with a poster size of 2 by 2 sheets and the results were impressive.

## CONCLUSION

Well, I'm a convert. *TurboPrint* hasn't increased printing speed but the difference in quality is unbelievable.

If you use your printer just for letters, there's no point in buying *TurboPrint*. It's really geared towards the printing of pictures, which it does very well. If you think your output could be improved upon, rush out and buy it now. **AS**

## SHOPPING LIST

*TurboPrint Professional* ..... £49.95  
by IrseeSoft, Am Schlachtbichel 1  
D-8951 Irsee, Germany

Distributed in UK by:  
HB Marketing  
☎ 0753 686000

## CHECKOUT TURBOPRINT PRO

### Features

Print quality plus a poster module, hardcopy options and even an IFF screen grabber! What more could you ask for?

### Ease of Use

*TurboPrint* is completely transparent!

### Documentation

The translation's a bit dodgy at times, but the manual is comprehensive.

### Print Quality

Even on my naff 24-pin, the results I obtained were quite simply astounding.

### Speed

Printing certainly isn't any faster under *TurboPrint*, but then it isn't slower either!

### Overall rating

A must for anyone who uses their printer for graphics. Miss it at your peril.

years ago, which never really achieved the recognition it deserved due to some rather lousy marketing this side of the Channel. Thankfully IrseeSoft has got its corporate act together and *TurboPrint Pro* is the result – the packing is slicker, the manual is more readable and the program's not bad either!

## PROTECTION RACKET

*TurboPrint* can be installed on both hard disks and boot disks, but there's one major problem that proves to be a real pain – it's copy protected. Each time you run the program in a fresh machine, *TurboPrint* asks you to insert the original disk. If it likes what it finds, then it'll continue but if your original becomes corrupt, you're stuck. Thankfully this only has to be done once every time the machine is switched on, so it's not a problem if you use your Amiga for long periods.

Getting *TurboPrint* up and running isn't that complex once you've fought your way through the strange English translation of what was a German manual. Things are helped along by a well designed Preferences-like program called *TurboPrefs* which allows you to carry out printer related tasks like telling *TurboPrint* which printer you have and so on. There's a wide selection of printer drivers bundled with *TurboPrint* including

There's a page set aside specifically for *TurboPrint*'s extensive colour correction options, one for the print mode (B&W, grey scale, colour etc), another printer drivers, density etc plus a fourth screen which handles *TurboPrint*'s Poster and Hard Copy options (more on these later).

One aspect of *TurboPrint* which is particularly powerful is the control it gives you over the dithering pattern used in screen dumps. There is a total of 12, ranging from the pattern used by the Workbench drivers to some more elaborate patterns such as vertical and diagonal lines, dots of varying sizes (a bit like the pattern used in newspaper print) and the usual Floyd-Steinberg dither modes.

## LOOPY PRINTS

The first thing you'll want to do once *TurboPrint* has been installed is to print something. Due to the design *TurboPrint* should work with all software without you having to inform the software of its existence. I tested it with a selection of titles and this claim certainly seems to be true.

The name *TurboPrint* is a little confusing. With the word 'Turbo' in it, you'd expect a speed improvement, but this didn't seem to be the case. The difference between the standard drivers and *TurboPrint*'s wasn't really worth writing home about. The only real difference you'll notice is in





# Fishing for Compliments

"There's loads of PD and shareware available for the Amiga - it's my job to find the best available."

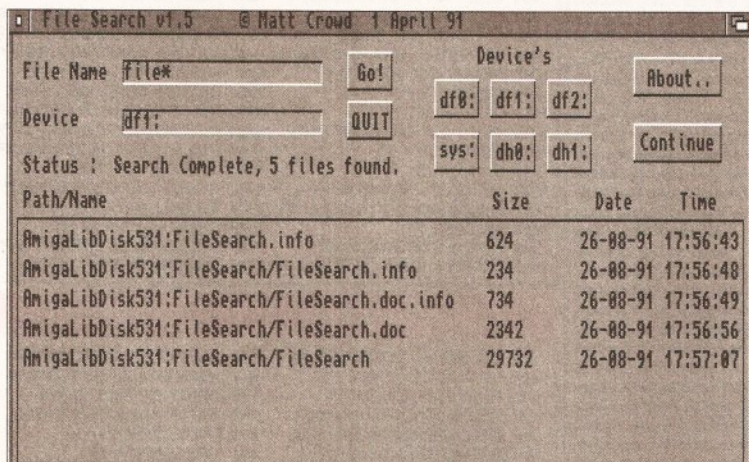
Ian Wrigley

This month, I'm mainly looking at programs available on recent Fred Fish disks. So rather than giving a 'value for money', I rate the individual program. You'll see from the table, which is on page 165, exactly which disks the programs are on - sometimes more than one reviewed program is from the same disk.

If you have access to a modem, many of the programs are available from bulletin boards such as CIX; if you don't, then the easiest way to get hold of the latest version of a program is to buy the Fish disk in question.

Often PD houses take the programs from the Fish disks and collect them into sets - for example, programmers' utilities, or 'business productivity packs' or whatever. However, it takes a little while for updates of programs to get to these collections, so your surest way of

**Ian Wrigley goes in search of more perfect PD, superb shareware... and fish?**



If you can never find the file you've been looking for, *Filesearch* may be the answer to all your problems. Be warned though - it's not exactly the fastest program in the world...

making sure that you have the latest version is to buy the Fish disk.

All the Fish disks mentioned this month were supplied by Unique Computing, which claims to be one of the most up-to-date sources of Fish disks in this country.

## BEGINNERS

What are PD and shareware?

PD stands for public domain, and is a term used to describe software which - quite simply - you don't have to pay for. It's free.

The only money you'll need to pay is to a 'PD house' which duplicates the disks and sends them to you.

Shareware is similar, except that if you like a program you are asked to send the author a fee for using it - normally around \$15-\$25.

It's a sort of 'try before you buy' scheme. If you don't like the program, you just delete it and you don't have to pay the author anything at all.

## BEGINNERS START HERE

## BEGINNERS

So where exactly can I get hold of this stuff?

PD and shareware is known as 'freely distributable software' - so if your mate's got a PD or shareware program that you like, you can just take a copy of his. Don't forget to send the author his fee if it's shareware, though. You can also get freely distributable software from 'PD houses', which make a small charge for duplicating disks. There's a list at the end of this article, and loads of adverts dotted about the magazine. Finally, if you've got a modem you'll find lots of PD and shareware on bulletin boards. And it won't cost you a penny to obtain (other than your phone bill, of course).

in place when the requester appears) and for the device to search - the program defaults to dh0: when launched, but there are buttons for a number of frequently-used devices. Hitting the 'GO!' button sends the program off to search for files - and that's where the only problem appears. It's sloooooow. OK, not as slow as searching through a 52Mb hard disk 'by hand', but even so you could take a few turns around the room before the program comes up with the goods.

Another criticism is that after *Filesearch* has found eight files which match the search criteria, it pauses and waits for you to hit the 'continue' button before it finds the next eight - wiping the first eight from the screen as it does so. What's wrong with the system found on a number of other computers, like the Mac, where the files are listed in a scrolling window which can be perused at leisure?

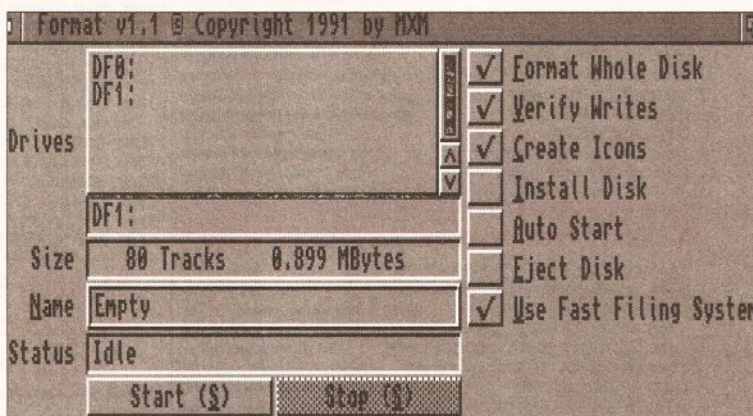
Still, beggars can't be choosers, and *Filesearch* does perform a function which is sorely missing from the Amiga.

Program rating .....6/10

## FORMAT

### Fish disk 535

Format 1.1 is a public domain utility written by Olaf 'Olsen' Barthel to replace the standard Workbench 2.x



For those not happy with the Shell, this replacement for Commodore's 'format' command enables you to do everything from the Workbench. *Format* produces a requester which lists all mounted devices with handy check boxes

## FILESEARCH

### Fish disk 531

This program, written by Matthew Crowd, is public domain. It enables a user to search any mounted volume - ram:, rad:, df0:, dh1: or whatever - for a specified filename. Although Amiga wildcards are not supported, the MS-DOS '\*' wildcard is ('\*' is the equivalent of '#?' on the Amiga).

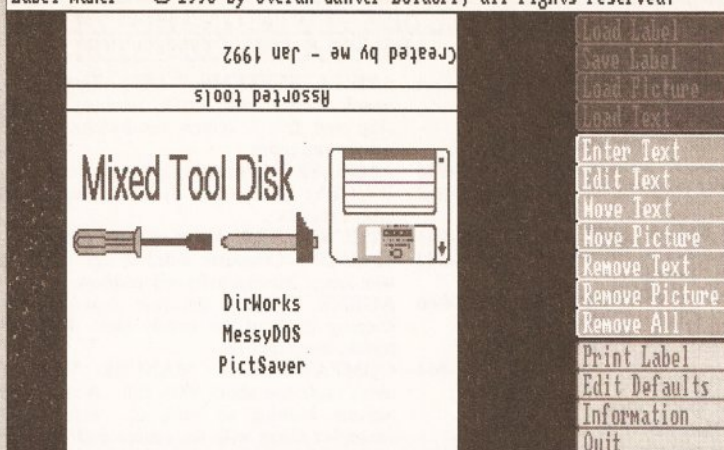
*Filesearch* can be launched from the Shell or directly from the Workbench, and presents a simple requester which asks for the filename (a '\*' wildcard is already

'format' command.

In the (extremely sparse) documentation he treats it as almost not worth talking about, but in fact it will be extremely useful to those Amiga users who aren't really too happy just typing 'format' from the Shell prompt, or for those who want access to a fully-featured disk formatting command from the Workbench (it can be accessed either way).

Running the program produces a requester which provides a list of all mounted devices, and a number of check-boxes for such

Label Maker ©1990 by Stefan Günter Boldorf, all rights reserved.



If your floppy disks are covered with crossed-out scrawl, **LabelMaker** could be the program you need to eliminate scruffy unreadable labels



**Filer** – yet another file management utility, but the screen layout is neat and uncluttered, making this one of the easiest to use

things as whether to use the FFS (fast filing system), whether to verify writes, whether to create icons etc.

These check-boxes can be activated either by clicking on them or just pressing a key on the keyboard corresponding to the first letter of the description. Give the disk a name (which defaults to 'empty') and you're ready to go.

**Format** is an extremely useful command – unless you're a die-hard CLI user, I recommend that you replace the format command on your Workbench disk working copy with this one immediately. It can even be made resident, if you so wish.

Program rating .....9/10

## POWERSNAP 1.1

**Fish disk 542**

Nico Francois' freeware **PowerSnap** utility gives Amiga users full 'copy and paste' functionality in all programs. It can be invoked from the Shell or from the Workbench, and supports a huge number of different options.

After installing the program (you

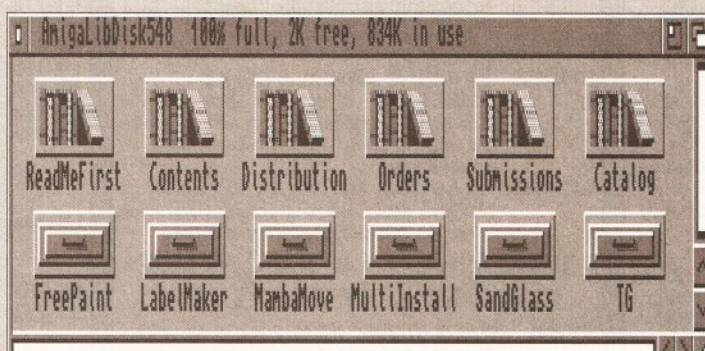
**"Term's a giftware program by Olaf Barthel – which means if you like it you send a gift."**

can either use the installation routine supplied or just double-click on the program's icon to try it out), holding down the left [Alt] key and clicking the left mouse button will mark the position at which to start copying text. Drag the cursor to the end position and the selected text will be highlighted.

To paste, you just move the cursor to the appropriate position

**Fred Fish** is an American who seems to spend all of his time collecting Amiga PD and shareware. Since its beginnings in 1985, the **Fred Fish** collection, as it's known, has grown to 580 or so disks of programs and utilities – that is, there were 580 disks available when this column was put together; by the time you read it there will doubtless be 20 or 30 more. Indeed, as the Amiga becomes more and more popular, so the rate of production of Fish disks increases.

## Who on earth is Fred Fish?



If you've got a **Fred Fish** disk in your collection you'll be able to recognise it immediately because all these disks have a standard appearance

Since disk 421, Fish disks have been created on an Amiga running Workbench 2.0, and indeed many of the new programs being distributed now require Workbench 2.0 to run. Of course that doesn't leave Workbench 1.3 owners out in the cold, though – many of the programs on the disks will still run on older machines. However, since machines sold in the States have been supplied with Workbench 2.0 for some time, and since Commodore has now dropped the A500 in favour of the A500 Plus, it is a fact of life that more and more programs will require the new operating system in the future. You should find that, wherever possible, I've noted whether the program will work with Workbench 1.3 machines. They have all been tested on an A500 Plus, running Workbench 2.04.

Unlike some PD distributors, Fred makes sure that his disks have a consistent look to them, so you don't have to waste time working out how to read the doc files or whatever. There is a standard line of icons containing information such as the contents of the disk, where to contact Fred and so on, and all documentation is set up to use the *MuchMore* text reader, which is included on each disk. The disks don't self-boot – you need to load Workbench to access them – but they include utilities like *LHarc* if it's necessary to use a program, so they're fairly self-contained.

and either hit [Alt]-T or press [Alt] and the right mouse button. The text is then copied into place. This copying isn't always instantaneous – it depends on the amount of text you've copied and the font that the text is in – but it's far faster than even the best touch-typist could manage.

**PowerSnap** has numerous 'switches' which make the program perform differently; these can either be entered in the normal way following the program's name if you are running it from the Shell, or by adding tooltypes from the icon's 'Info' window if you are running it from the Workbench. These include selecting what key should be pressed in conjunction with the mouse button for copying and pasting, adding text before and after the selection is pasted, and removing all carriage return characters, so that the pasted text is run on to one line.

**PowerSnap** recognises all Amiga fonts, including scalable fonts in Workbench 2.0, and fully supports clipboard.device, so you'll be able to paste into programs which support that either by using **PowerSnap** or the program's own Paste command.

The program is compatible with Workbench 1.2, 1.3 and 2.0, and is highly recommended – it's another that I will be installing and using all the time.

Program rating .....8/10

## POWERPACKER UTILITIES

**Fish disk 542**

Fish disk 542 contains routines which support Nico Francois' **PowerPacker** program, which is used to crunch files. *PPMore* is the main program – it allows you to read ASCII files and files which have been crunched with **PowerPacker**.

Running it (from Shell or Workbench) produces a requester

continued on page 161













# AMIGA SHOPPER

**SPECIAL OFFER**

## Start programming with our special offers

### AMOS

**SAVE  
£14**

AMOS is the best-selling version of BASIC for the Amiga. It offers powerful graphics commands and an easy-to-use editor, allowing you to create powerful programs simply. There are many AMOS programs in the public domain to learn from, plus our monthly column.

**Only £35.95**

### DEVPAK 3

For achieving sheer speed on the Amiga, you need to use machine code. And there's no better way to generate it than using Hisoft's acclaimed Devpac 3 assembler. An absolutely vital utility for programmers everywhere.

**Only £65.95**

**NEW  
VERSION**

### AMOS COMPILER

**SPEED  
FOR  
£21.95**

For those occasions when AMOS simply isn't fast enough, the AMOS Compiler transforms your programs into speed fiends! Your existing AMOS programs will run without alteration at a considerably faster rate. Essential for serious work.

**Only £21.95**

### HISOFT PASCAL

Programmers have been anxiously waiting for this – Hisoft's version of the popular Pascal language. An extremely powerful language, it is easier to learn than C while offering many of the same facilities. An ideal choice when upgrading from BASIC.

**Only £89.95**

**SAVE  
£10**

Please send me the following programming products:

AMOS (£35.95) ☐ AS012      Devpac 3 (£65.95) ☐ AS014

AMOS Compiler (£21.95) ☐ AS013

Hisoft Pascal (£89.95) ☐ AS105

Total: £ .....

Method of payment (please ring): Visa Mastercard Cheque PO

Make all cheques payable to **Future Publishing Ltd**

Card No: \_\_\_\_\_ Exp date: \_\_\_\_\_

Name .....

Address .....

.....

.....

Postcode .....

**Send to:** Future Publishing, The Old Barn, Somerton, Somerset  
TA11 7BR

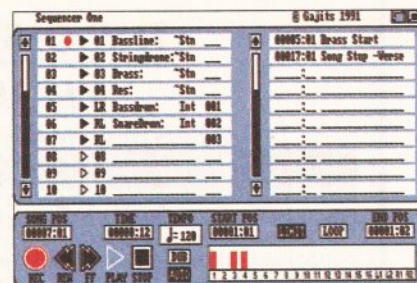




## Sequencer One

Gajits RRP £89.95 runs in 512K

Probably the best entry-level sequencer there is. It's a simple and accessible way of writing your own tunes, whether you use a MIDI keyboard or just the Amiga's sound capabilities. Comes with a whole set of samples to get you started and demo tunes which you can take apart to see how it works.



"All you'll need at first and none of the frills that a beginner won't use... one of the most straightforward sequencers we've seen" **Amiga Format Issue 24 80%**

## The Graphics Workshop

Holosoftware Technologies RRP £79.95 runs in 1Mb or more

The superb paint and animation program that takes Amiga art on from Deluxe Paint 3. Features an enormous range of painting tools implemented in an intelligent, easy and accessible way, plus numerous unique ideas: multiple palettes; 'objects'; and a phenomenal range of animation techniques.



"Most animation packages fall down on ease of use: not Graphics Workshop... there are many advantages over Deluxe Paint" **Amiga Format Issue 25 Gold 90%**

# What would you say to £170 worth of top graphics and music software for only £5?

(PLUS FREE GIFT: the biggest and most exciting issue ever of the world's best-selling Amiga magazine!)

### Amiga Format

is the world's biggest-selling Amiga magazine with over 130,000 readers every month. It's recognised by the computer trade as the magazine that gives people the best advice about what to buy. It's the Amiga magazine that covers everything Amiga, from the serious to the seriously fun. And this Christmas bumper special edition is the biggest Amiga mag ever printed, with 308 pages of interesting articles and useful advice.

### Sound and Vision Supplement

contains complete instructions for getting started with your music and graphics software, as well as an introduction to the world of computer music and a brief explanation of the diverse world of Amiga animation.

### Christmas Lemmings

#### Psygnosis

So the festive season is past, but you'll still be enchanted by this playable demo of three unique new Lemmings levels put together specially for Amiga Format by DMA Design. If you've never played the game, find out what all the fuss is about!

## Yes please!

I would like to purchase \_\_\_\_\_ copy/ies of Amiga Format issue 30 **Plus** Sound and Vision supplement, Sequencer One and Graphics Workshop. at a cost of £5 per copy (including P&P)

Total cost £ \_\_\_\_\_

Method of payment  
Cheque ☐

Credit card ☐  
Card number   
Expiry date

Name   
Telephone   
Address

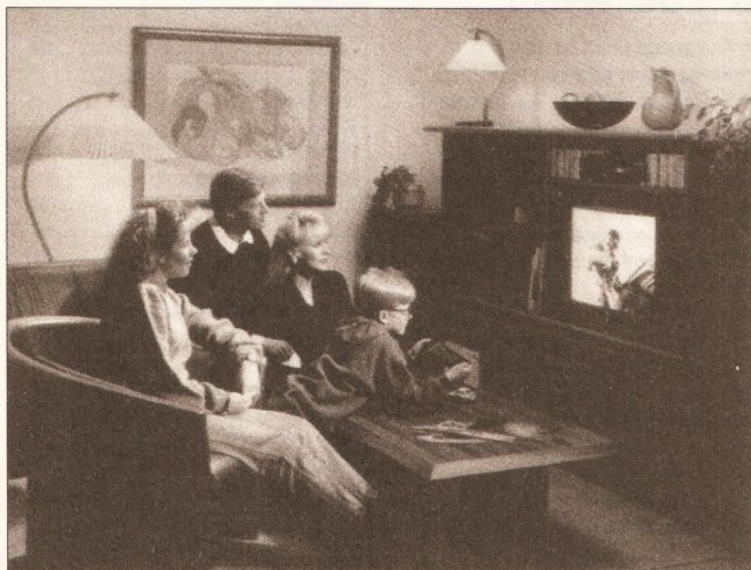
Post Code

ENCLOSE THIS COUPON (WITH CHEQUE IF APPLICABLE) IN AN ENVELOPE TO:  
**AMIGA FORMAT SUBSCRIPTIONS, FREEPOST, SOMERTON TA11 7BR**

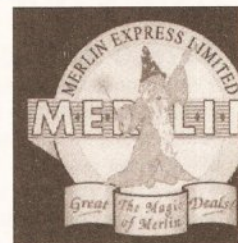


WIN • WIN • WIN • WIN • WIN • WIN • WIN • WIN

# Win a CDTV and join the info revolution!



**Conjure yourself a magical CDTV from those wonderful Merlin people by answering our three easy questions**



**M**ulti-media is certainly the happening thing at the moment, and now *Amiga Shopper* along with Merlin Express are offering you the chance to beat the Joneses and be the very first in your street to get

in on the act. Yes, that's right, you could be the proud owner of one of Commodore's Amiga CDTVs (formerly CDTV)!

CDTV, for those of you who don't know already, stands for Commodore Dynamic Total Vision, and it's the

company's attempt to bring multi-media into every home.

But what, you may ask, do we mean by 'multi-media'? By 'multi-media' we mean lots of whizzo pictures, animations, sounds, speech and text all in the same application. Imagine an encyclopedia that can talk and show clips of film. Imagine being able to cross reference a vast store of information at the click of an infra-red remote control hand set. Imagine having a CD with 650Mb (that's more than 750 ordinary floppies) of information on a single subject.

We are moving into an era of history in which information will be the most valuable commodity. CDTV gives the home user easy access to this information, and all through an attractive user interface.

Obviously CDTV is invaluable for reference, but its possibilities for education are truly awesome. Check out Mark Smiddy's article on page 15 for details of the stunning packages already available. As Mark concluded, "I have seen the future of education: its name is Commodore Dynamic Total Vision."

So if you'd like the chance to try out one of these amazing machines for yourself, then all you have to do

is have a go at the questions below. Jot down your answers on a postcard or the back of a sealed envelope and send them to us at:  
CDTV Competition  
*Amiga Shopper*  
29 Monmouth Street  
Bath BA1 2DL  
The closing date is 10 April.

## THE CHALLENGE

### QUESTION 1

**Whereabouts in Cornwall is Camelot believed to have been?**

- a) Porthcurno
- b) Polperro
- c) Tintagel

### QUESTION 2

**What was the name of King Arthur's court magician?**

- a) Paul Daniels
- b) Merlin
- c) Faust

### QUESTION 3

**Who was King Arthur's wife?**

- a) Gwendolin
- b) Guinevere
- c) Guatemala

## ARE YOU ONE OF OUR WINNERS?

A couple of months back an Amiga 500 Plus was the star prize on offer from Gordon Harwood Computers. The answers to that competition way back in December go like this:

- 1) It was Andy Warhol who helped launch the original Amiga.
- 2) The Agnus chip on the Amiga 500 Plus can address 2Mb of Chip RAM.
- 3) HAM stands for Hold and Modify.

After much deliberation and sorting through the flood of entries – by far the biggest response yet – we have come up with a winner. It is...  
LY Songyow of South Norwood, London.

## TWO SHORT YEARS AGO...

A brief look at the Amiga's shady past...

- The popular PD disk, *The Amiga C Manual*, was released by Anders Bjerin and came to the rescue of many would-be programmers.
- An Amiga-based system was launched for training air traffic controllers. It featured a 68020

processor running AmigaDOS and a graphics card giving a resolution of 1,280 x 1,024 pixels on a touch sensitive screen.

- Commodore launched the *Flight Of Fantasy* bundle.
- The stunning Mandala virtual reality system, based around the Amiga, made its first appearance.

## ...AND NEXT MONTH

But seriously folks, what you really want to know is when the next *Amiga Shopper* hits the streets – and who can blame you?

We'll be back on 5th March, and will be covering:

- Hard Drives – at last Jolyon Ralph explain's everything you ever wanted to know and weren't particularly afraid to ask.

● The amazing DCTV graphics system gets the in-depth AS treatment from Gary Whiteley and Jason Holborn.

- Paul Overaa takes a look at *Devpac 3*, the latest incarnation of HiSoft's acclaimed assembler.
- *Amiga Shopper* – buy it, you know it makes serious sense.

# Mavis Beacon Teaches Typing

**-simply the finest typing instructor in the world.**

Learn from the best with this award winning, best selling program. This expert system takes you step-by-step through the entire keyboard. Selecting quotes from history's greatest writings, using riddles and hundreds of fascinating facts from the Guinness Book of World Records\* this unique program makes learning interesting and fun.

Mavis Beacon Teaches Typing creates customized learning exercises and gently coaches you along at your own speed and skill level, checking your progress lesson by lesson.

- Stunning 2D and 3D graphics including life-like keyboard, 'Guide-Hands' and on-screen progress graphs.
- Help windows and Speed, Accuracy and Completion meters all appear on-screen.
- Standard Querty or 'Dvorak' keyboard lessons, Word Processor or 'Typewriter' mode and practise texts as used by employment agencies included.
- Complete typing textbook included.

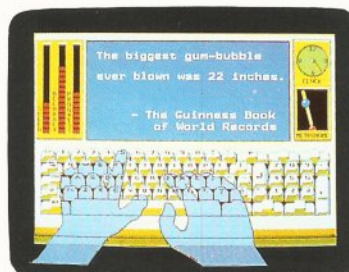
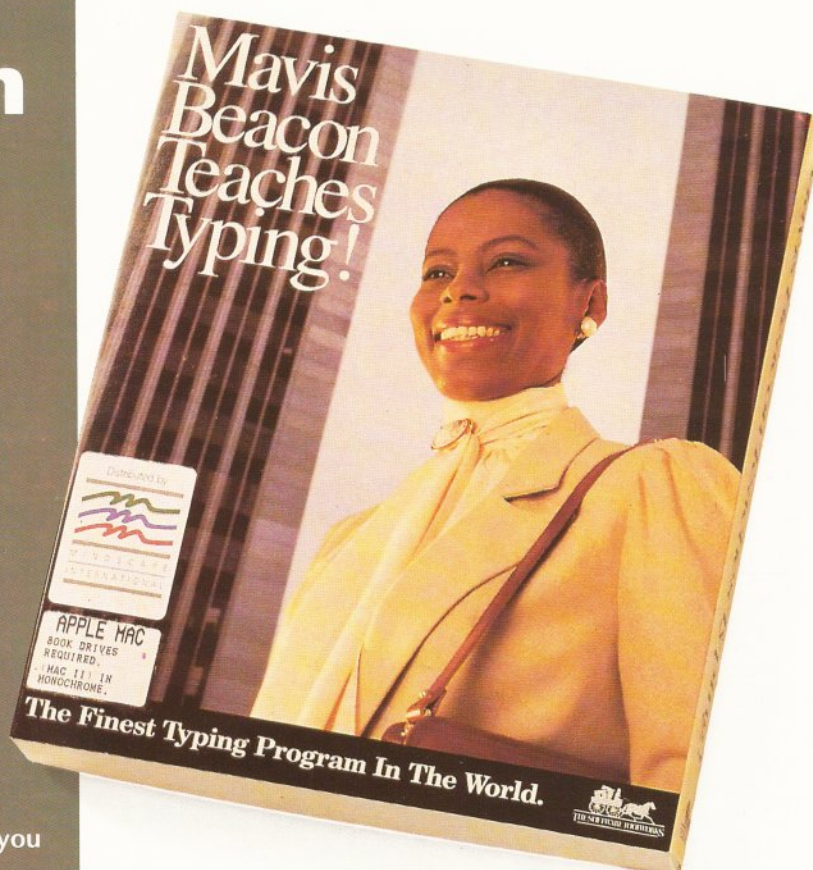


**THE SOFTWARE TOOLWORKS**

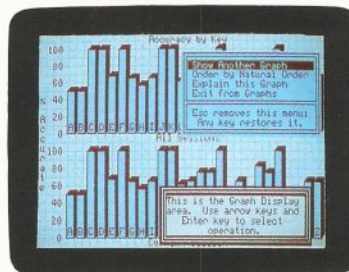
The Coach House,  
Hooklands Estate, Scaynes Hill,  
West Sussex RH177NG

Tel: Sales (0444) 831761  
Product info' (0898) 234214

\* Guinness Superlatives Ltd, published by Sterling Publishing Co, Inc. New York, N.Y.  
© Copyright 1990 The Software Toolworks, Inc.



'Guide Hands' type with you to show perfect technique.



Speed, Accuracy and Completion meters on-screen.

Name: \_\_\_\_\_

Address: \_\_\_\_\_

Please send me:

Formats:	Amiga	£30.99 inc VAT	<input type="checkbox"/> Tick
	PC Dual	£30.99 inc VAT	<input type="checkbox"/>
	Atari ST	£30.99 inc VAT	<input type="checkbox"/>
	Apple Mac	£30.99 inc VAT	<input type="checkbox"/>
	Commodore C64	£30.99 inc VAT	<input type="checkbox"/>

Payment: Access or Visa

Expiry Date:       ☐ Tick



Or by cheque made payable to:  
'Mindscape International Ltd'

☐ Tick

